We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

4,800
Open access books available

122,000
International authors and editors

135M
Downloads

Our authors are among the

154
Countries delivered to

TOP 1%
most cited scientists

12.2%
Contributors from top 500 universities

WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com

12

Analog-aware Schematic Synthesis

Yuping Wu
Institute of Microelectronics, Chinese Academy of Sciences, China

1. Introduction

An analog circuit has great requirements of constraints on circuit and layout optimization for the purpose of functionality. Various constraint generation methods were provided, but there are too many limitations even the circuit topology has a bit variance due to no knowledge of the circuit functionality. To get the requirements exactly, you must know the circuit functionality exactly before, so analog circuit functionality analysis is very important for analog circuit design, especially for automatic analog/mixed signal design, but until now there is few method research report for automatic analog circuit functionality analysis except for the digital system design. The conventional way is that most of the work is done from an analog structural feature highlighted circuit schematic by the engineer manually, that is to say a good circuit schematic is the precondition for manual analysis on circuit functionality, which brings another issue about analog circuit schematic generation for analog / mixed signal design automation.

It should be appreciated that the circuit schematic generation has been in use for years with digital designs, functional clustering based analog circuit schematic generation was reported in [37, 39-43], which is rule-based and only feasible for some simple functional blocks due to the limitation of the description of rules. In the commercial tools from Cadence, Synopsys, and Magma, they use the methods from digital [8] for analog as instead, user cannot get the analog structural features insight, so it is hard to get the constraints for circuit and layout optimization from the schematic, although some previous works have been done [9][44].

In the long term, analog schematic generation is also necessary for future analog synthesis and analog design migration. The complete analog design automation flow is a far-away perfect expectation, as the part of such synthesis flow, analog behavioral synthesis will transform the behavioral description into circuit netlist, and the circuit netlist will be transformed into analog schematic, also such analog-aware schematic synthesis is the technical base to schematic optimization / retuning for analog design technology migration.

To overcome such issues, we studied a structural feature-based analog circuit analysis and partition technique, generated the constraints for schematic generation, circuit optimization and layout optimization after circuit analysis; based on that, we proposed an algorithm to generate analog aware circuit schematic [12] from the partitioning results with analog functionality and structural features highlighted, the constraints for circuit and layout optimization are identified on that schematic, and also analog functionality and structural feature can be got insight intuitively, which is helpful to circuit designers and layout engineers for circuit optimization and layout optimization.

This chapter describes the implementation of such analog-aware circuit schematic synthesis, and is organized as: section 1 gives the technical background necessitates for analog-aware circuit schematic synthesis; section 2 will present the analog-aware schematic synthesis flow; section 3 will detail structure features of analog functional circuits and descriptions, which includes low level analog structure features, high level analog structure features, structure feature library composition, structure feature associated attributes, and structure feature recognition; section 4 will describe analog circuit functionality analysis and partitioning, which includes input information, pre-processing, tracing direct current paths, tracing signal paths, encoding for blocks, checking isomorphism and quasi-isomorphism, and partitioning into hierarchy; section 5 will describe the constraint generation, which includes constraints for schematic generation and optimization, constraints for circuit design and optimization, and constraints for layout design and optimization; section 6 will describe analog schematic generation, which includes the symbol generation based on functionality, symbol placement, wiring, and constraint identification; section 7 will describe analog-aware schematic synthesis with companion circuits, which includes common feature extraction, functionality analysis and partitioning, constraint extraction with companion circuits, and analog schematic generation with companion circuits; and finally we will show some experimental results of such analog-aware circuit schematic synthesis technology.

2. Analog circuit schematic synthesis flow

As shown in Fig. 1(a), the traditional analog circuit schematic synthesis consists of 1) netlist-in; 2) data-in for mapping between devices and symbols; 3) cell symbol generation; 4) symbol placement for devices, cell instances, and ports; 5) wire routing; and 6) schematic-out. In comparison, the new analog circuit schematic synthesis flow consists of 1) netlist-in; 2) data-in for mapping between devices and symbols; 3) template-in for functionality analysis; 4) functionality analysis and partitioning for new hierarchy; 5) port analysis; 6) constraint generation; 7) analog-aware symbol generation; 8) analog-aware symbol placement; 9) analog-aware wire routing; 10) analog-aware constraint identification; and 11) schematic-out as shown in Fig. 1(b).

In the two schematic synthesis flows, as the common parts, circuit netlist-in can be spice-compatible netlist or netlist-in-database consisting of devices and connections; data-in for mapping between devices and symbols will set up one-to-one relation between devices and symbols for correct device symbol use; and schematic-out pushes the schematic data into the EDA platform database, such as DFII or OA, so that the schematic viewer/editor can display the schematic directly.

The differences between the traditional flow and the new flow are in red color. The first difference between them is the introducing of the templates-in. The templates-in includes circuit templates, symbol templates, and constraint templates. A circuit template has a couple of associated symbol templates and constraint templates.

Circuit templates are used for functionality analysis and partitioning with bottom unit circuit description and complex high level block composition description. The template for unit circuit must describe the device composition and connections of the unit circuit with transistor level in detail and stamp the functionality correctly; while the template for complex high level circuit must describe composition of sub-functionalities and connections among functional blocks, and also the functionality of the complex circuit must be stamped

with functionality name correctly. All the functionality names are used for functionality analysis of complex high level circuit based on the specified name conventions.

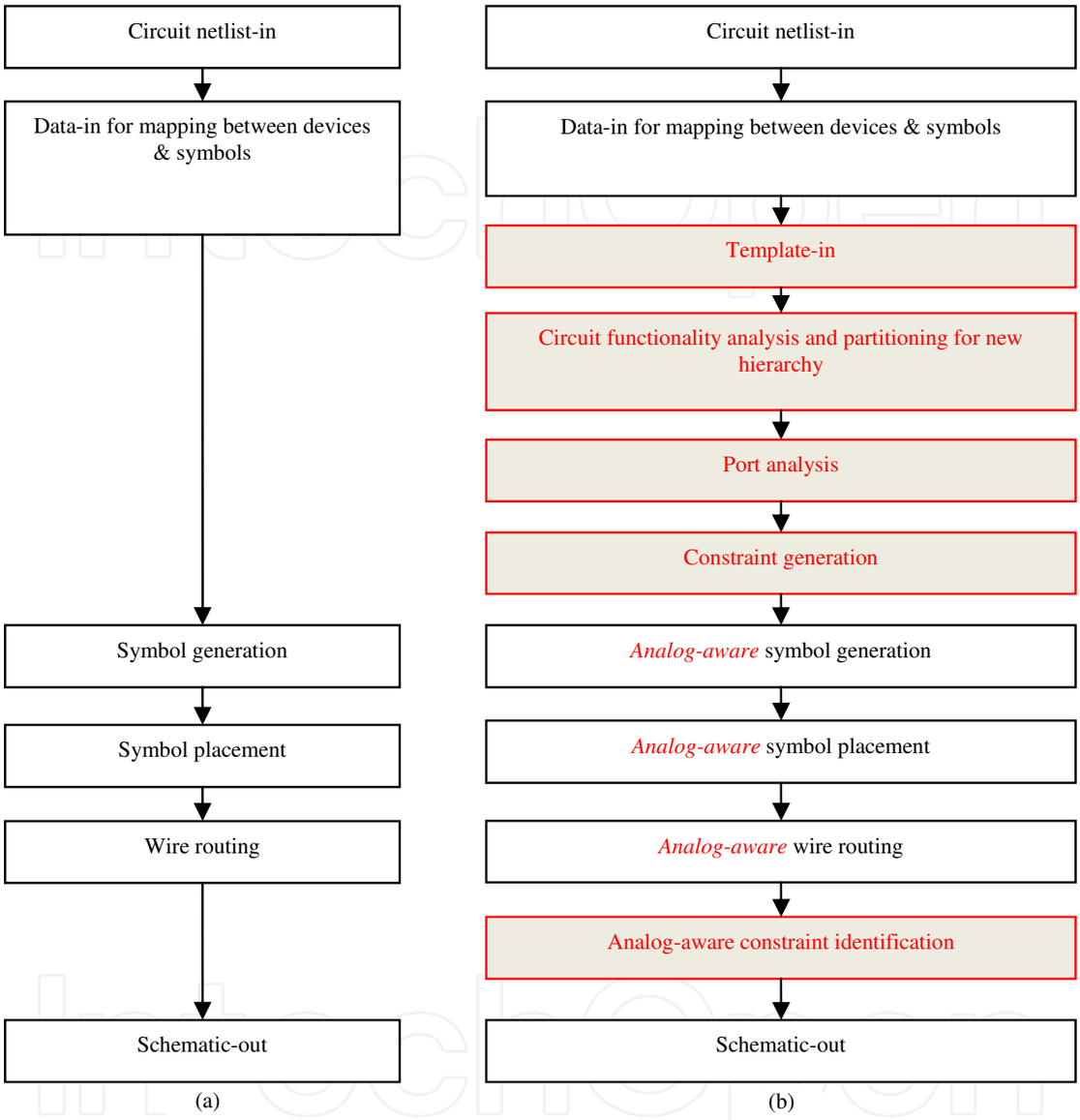


Fig. 1. Comparison of traditional analog circuit schematic synthesis flow (a) and novel analog circuit schematic synthesis flow (b)

Symbol templates are for symbol generation based on the functionality, designers can get functionality from the shapes of symbols, due to the symbol shape reflecting the functionality intuitively.

Constraint templates are for generating sizing, floorplanning, and layout constraints, which will speed up analog schematic synthesis, circuit sizing, floor-planning, and layout synthesis by reducing the possible exploration space and making the solution candidates more reasonable and acceptable [10]. The template for constraint generation can be built by designers manually or from good designs by automatic extraction tools.

The second difference between the flows is the introducing of analog circuit functionality analysis and partitioning for new hierarchy, which is the most solid base of the new flow and will be a bit detailed in next section.
The third difference between the flows is the introducing of port analysis. In traditional schematic synthesis flow, due to lacking of port analysis, all of the ports for each cell are treated as inputs/outputs no matter what they are in purpose exactly, so the synthesized schematic looks confused from the ports. Correct identification of port attribute is very important in schematic, so the port attribute should be captured before, but it is impossible to specify the port attributes manually for all the cells in a design especially when the design is in large scale, designers can only input some for several of them. Hence, it is necessary to use an automatic program to solve such issue. We introduce the port analysis for it, it determines the port types for each sub-cell automatically based on the combination of functionality partitioning, circuit template, signal flow analysis, dummy connection, ESD connection, substrate connection, name convention, and so on. The port analysis result will be used for pin placement on cell symbol generation and port terminal symbol selection and placement on analog-aware symbol placement step.
The fourth difference between the flows is the introducing of constraint generation for schematic synthesis, circuit sizing, floor-planning, and layout optimization, which is based on the combination of functionality partitioning, constraint templates, signal flow analysis, port analysis, dummy connection, ESD connection, MOSCAP connection, and so on. The constraints include symmetry requirements in a DC path, device matching requirements among DC paths, symmetry requirements between DC paths, dummy devices, protection devices and the associated protected devices, MOSCAP devices, critical signal nets, net current, and net wiring width, etc.
After analog-aware symbol placement and wire routing steps, as the fifth difference, analog constraint identification on the schematic is necessary to make circuit designers and layout engineers have a good insight on the design for circuit optimization, physical floor-planning, and layout optimization. The identifications include symmetry requirements in a DC path, device matching requirements among DC paths, symmetry requirements between DC paths, dummy devices, protection devices and the associated protected devices, MOSCAP devices, critical signal nets, net current and net wiring width, and so on. All the identification contents are results from the steps of functionality analysis and partitioning, port analysis, and constraint generation.
In summary, the great differences between traditional flow and novel flow are the introducing of template-in for functionality analysis, functionality analysis and partitioning for new hierarchy, port analysis, and constraint generation by the novel flow, which makes it possible for analog-aware symbol generation for cells, symbol placement, wire routing, and constraint identification on schematic based on the functionality, port types, and other constraints, so the innovation of the flow is the functionality analysis and partitioning technique, port analysis, automatic constraint generation, and constraint-driven analog-aware schematic generation.

3. Structure features of analog functional circuits and descriptions

Structure features of analog functional circuits are the intuitive bases for setting up the circuit templates directly and setting other associated constraint templates. The structure feature of analog functional circuits includes low level analog structures and high level analog

structures; the first focuses on the composition of devices and their connections, and the later focuses on the composition of basic or complex function blocks and their connections.

3.1 Low level analog structure features[1-3]

3.1.1 Structure features for basic amplifier circuits

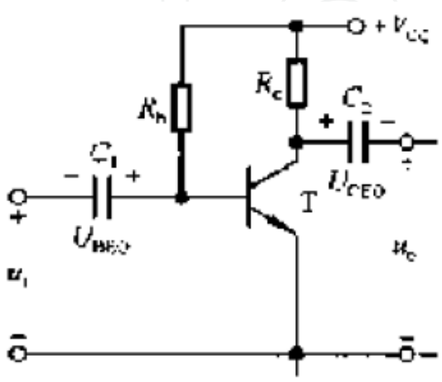


Fig. 2. Structure features for CE amplifier

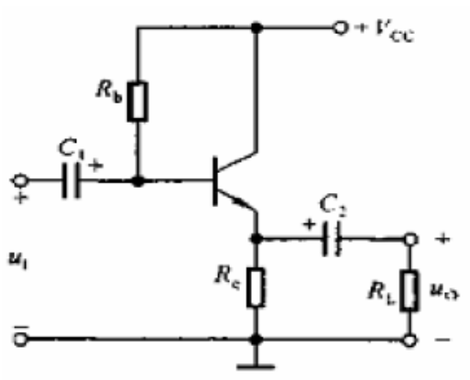


Fig. 3. Structure features for CC amplifier

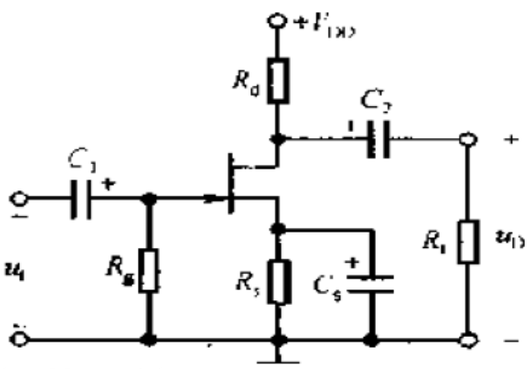


Fig. 4. Structure features for CS amplifier

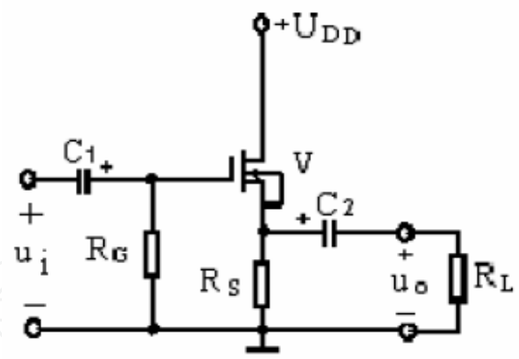
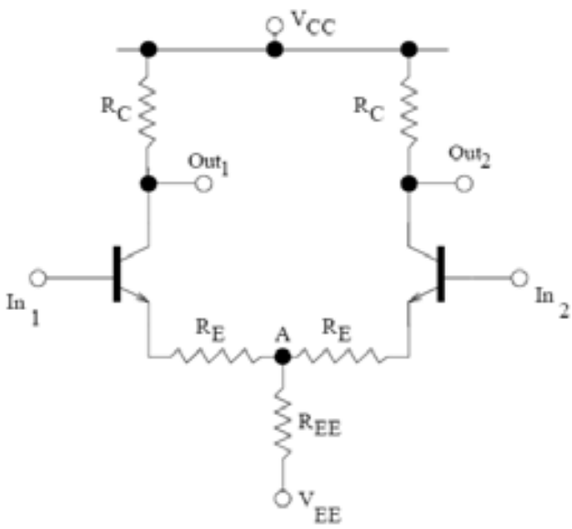
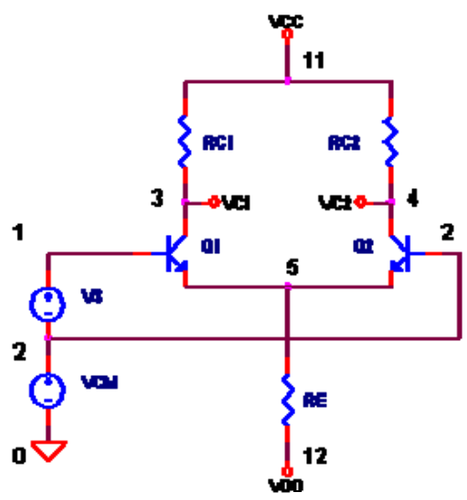
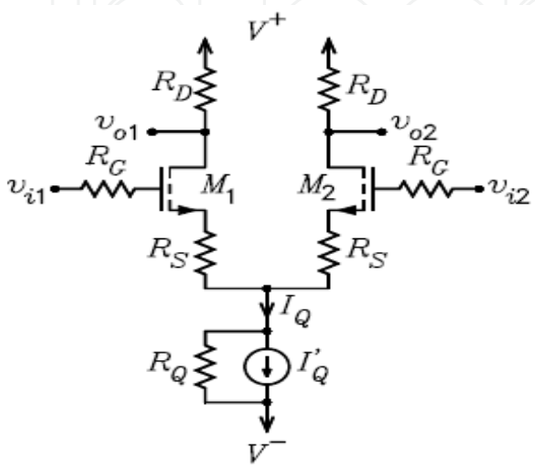
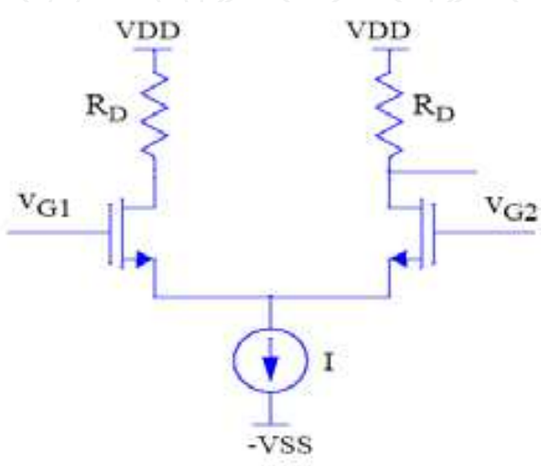


Fig. 5. Structure features for CS amplifier









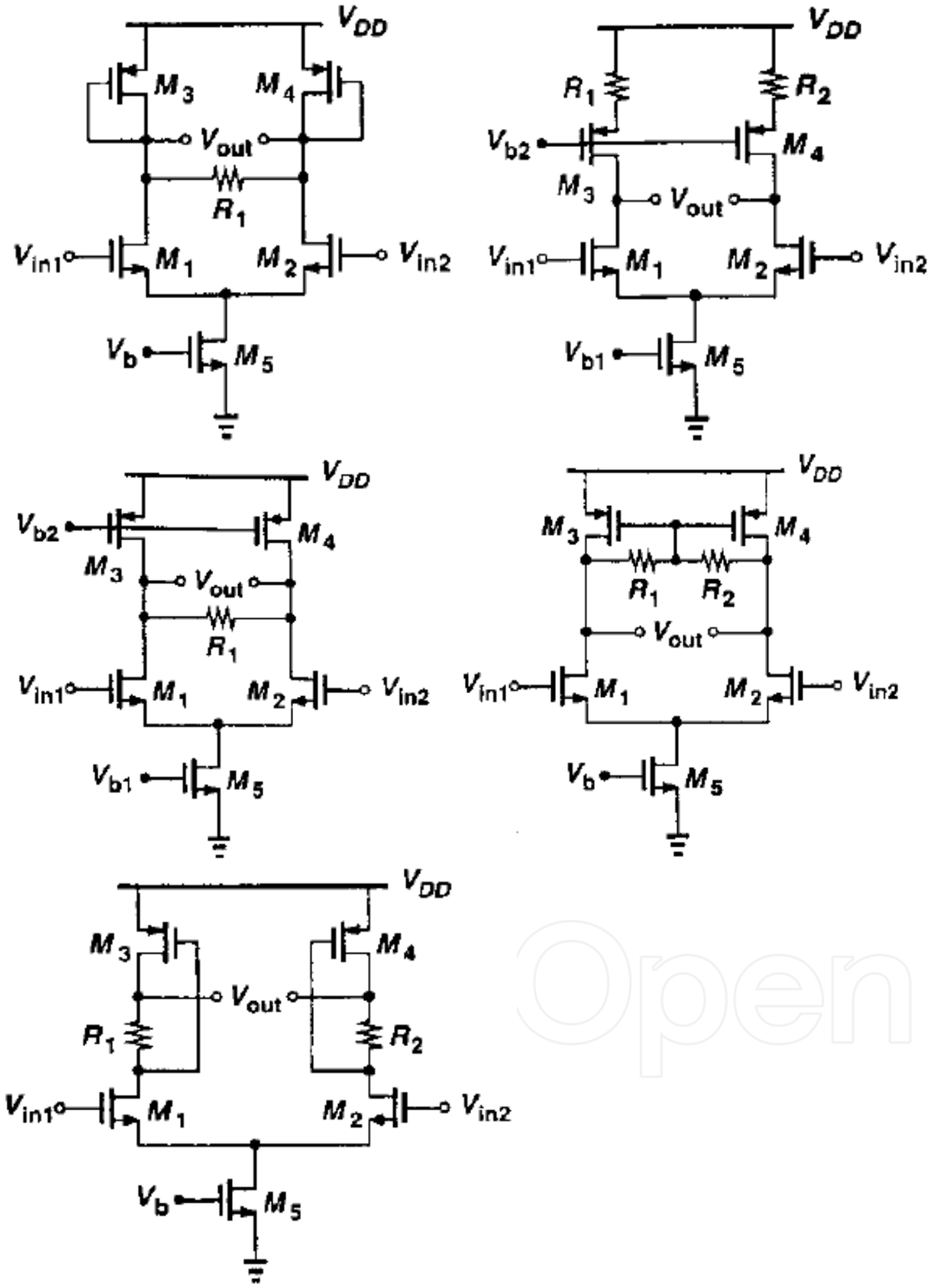


Fig. 6. Structure features for differential amplifiers

3.1.2 Structure features for amplifier output circuits

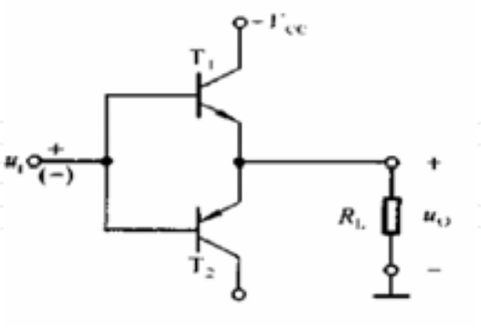


Fig. 7. Structure features for OTL circuit

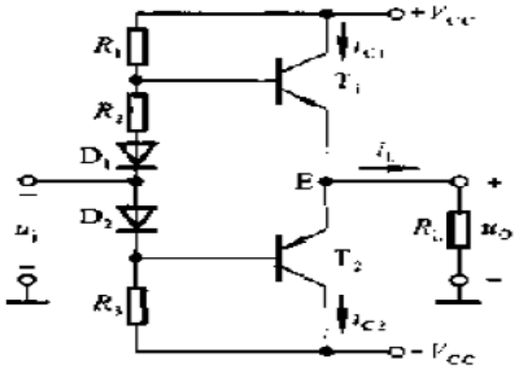


Fig. 8. Structure features for OCL circuit

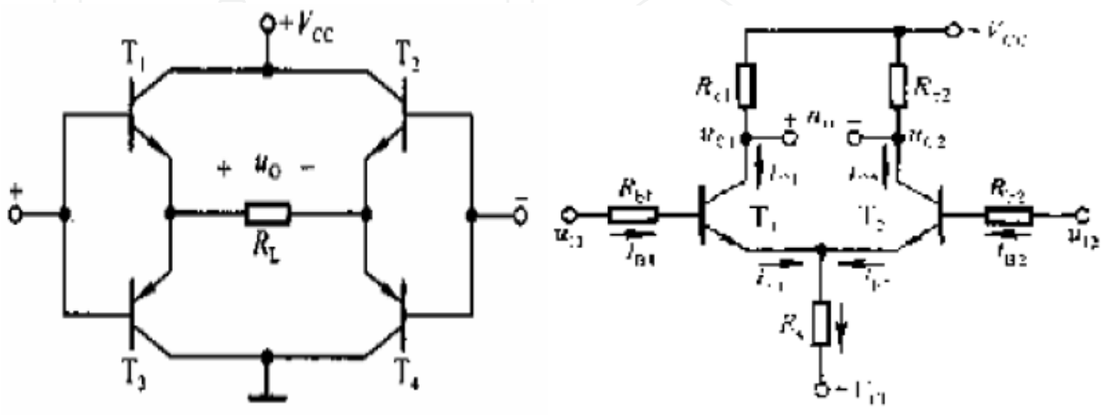


Fig. 9. Structure features for BTL circuits

3.1.3 Structure features for current source circuits

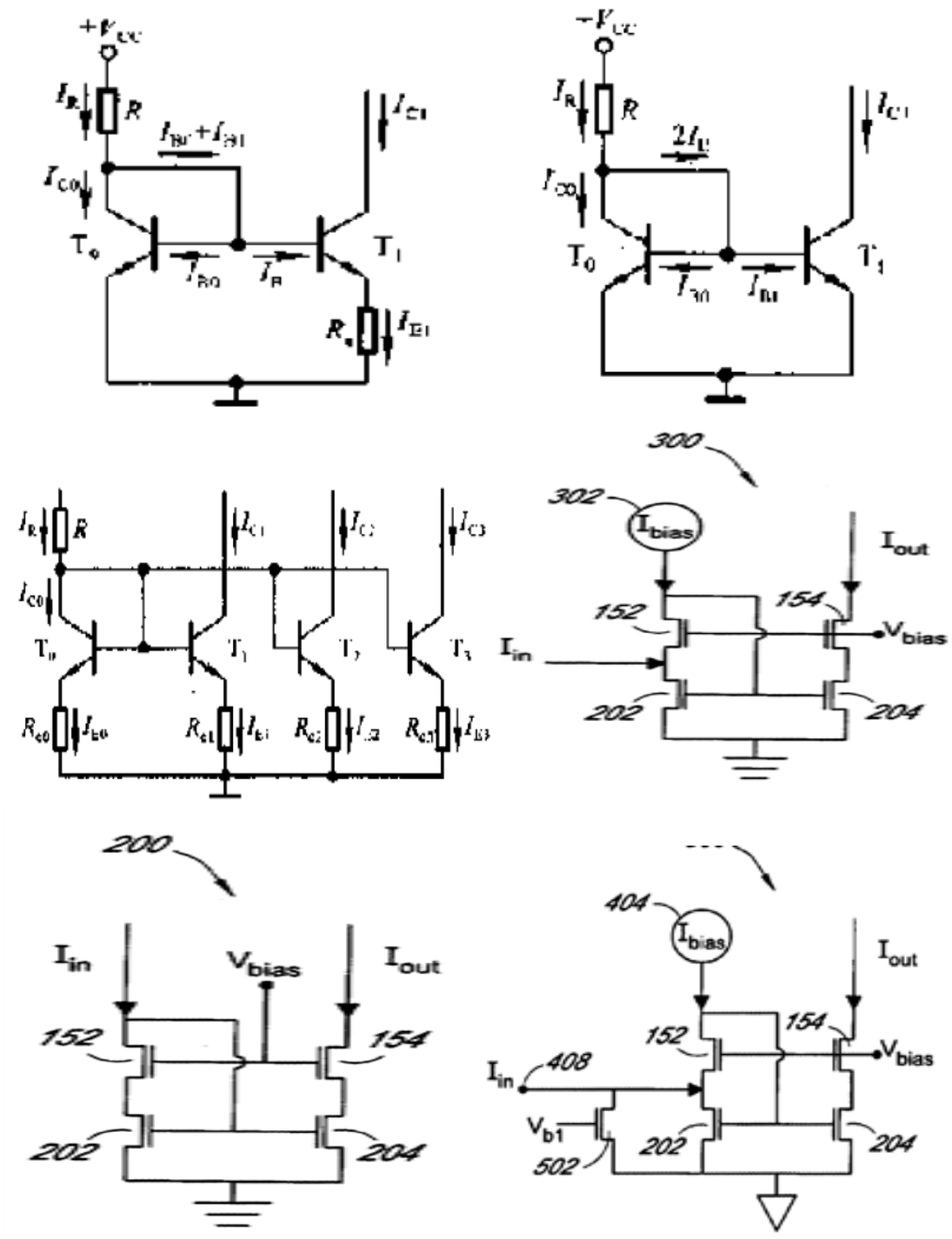


Fig. 10. Structure features for current mirror / current source circuits

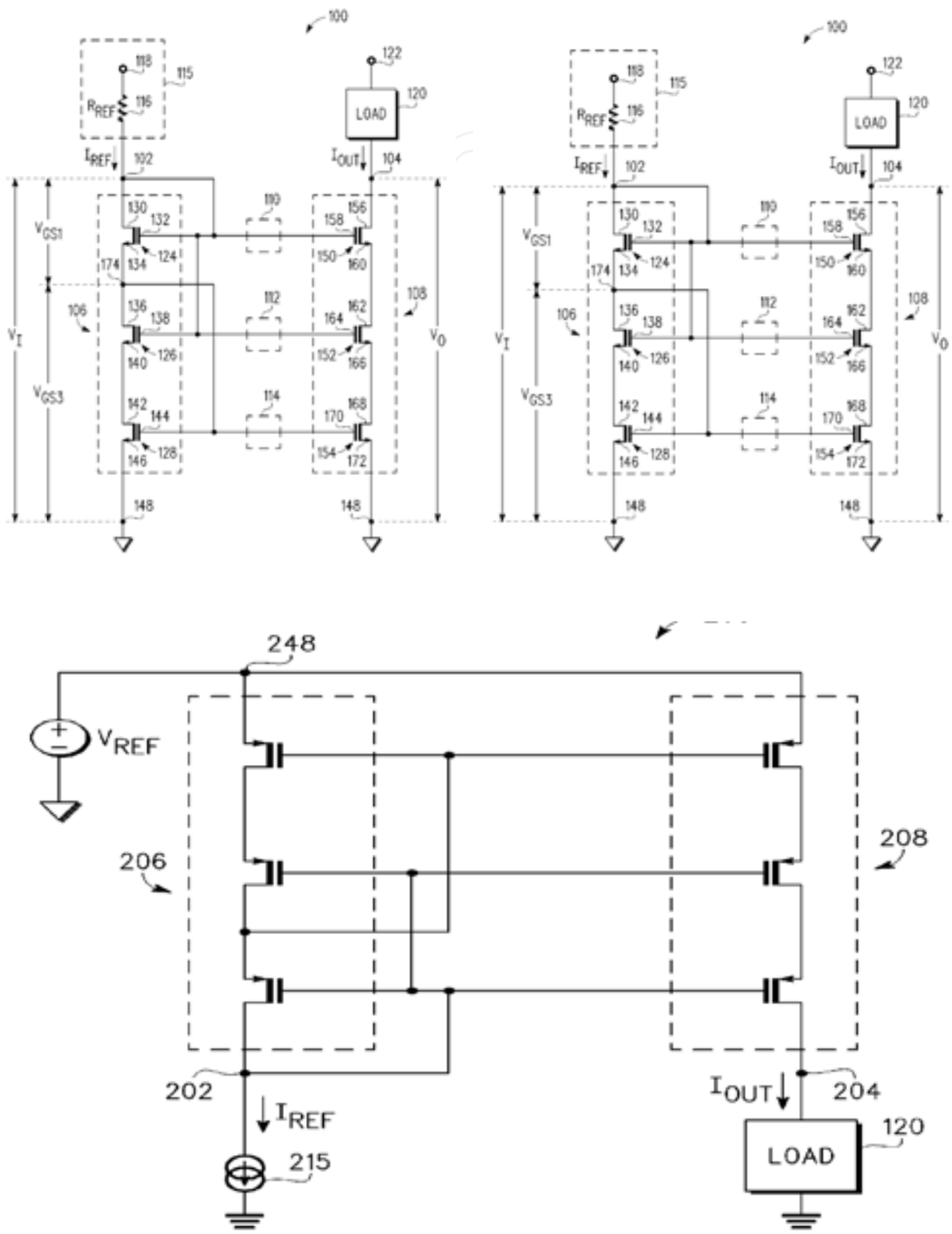
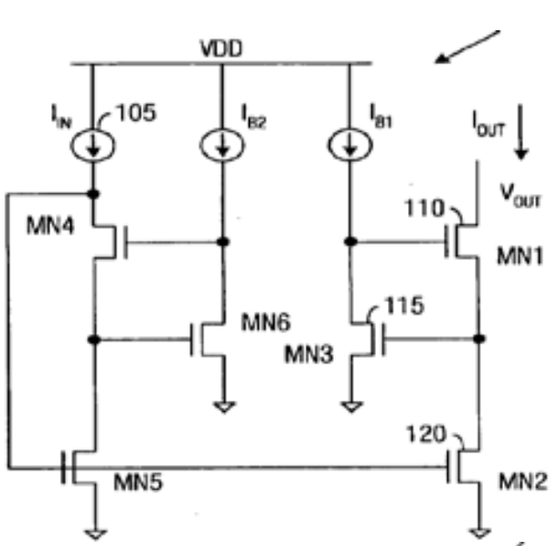
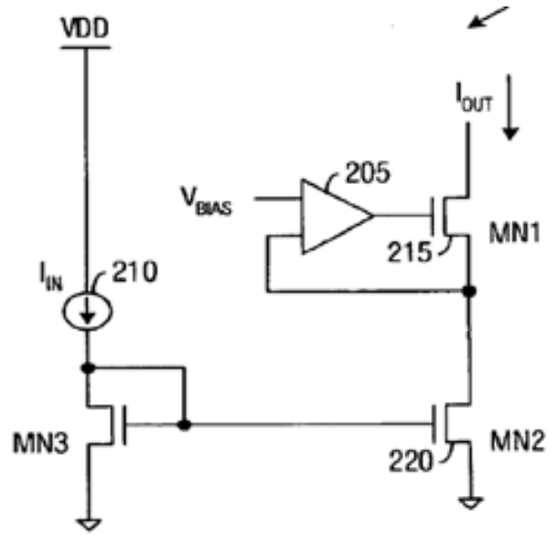
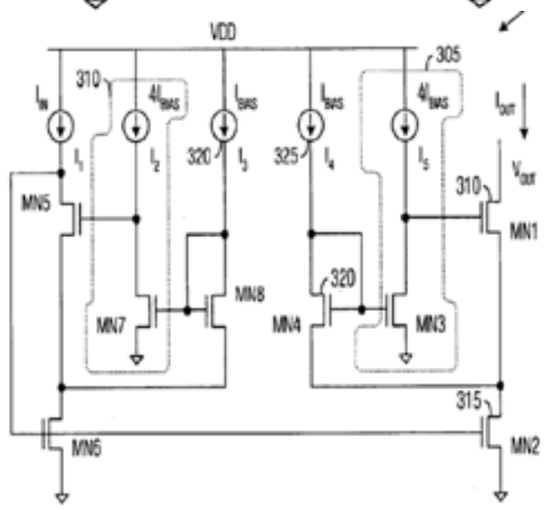
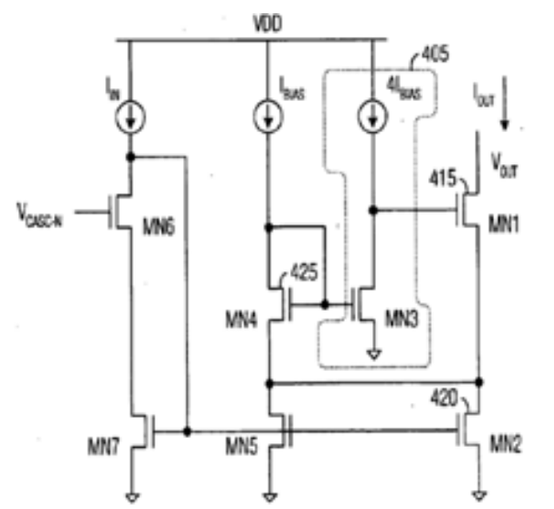


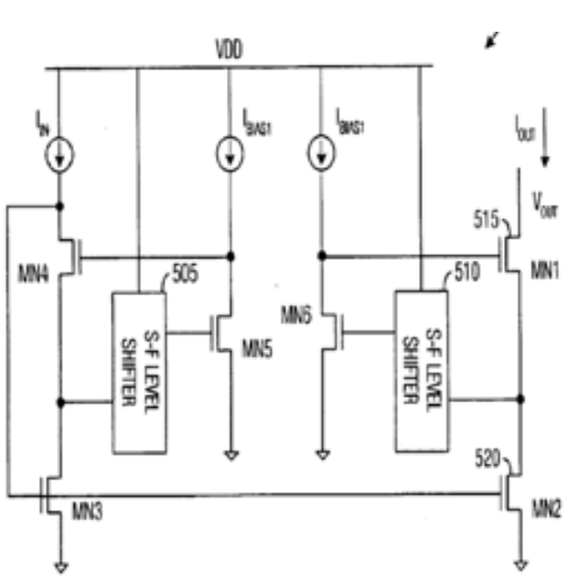
Fig. 11. Structure features for stack cascade current source circuits

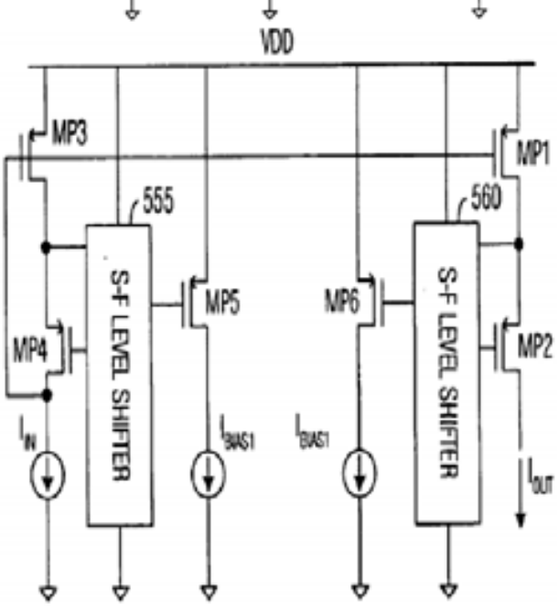

VDD
I_{IN}
105
I_{B2}
I_{B1}
I_{OUT}
V_{OUT}
110
MN4
MN1
MN6
115
MN3
120
MN5
MN2


VDD
I_{OUT}
205
V_{BIAS}
215
MN1
I_{IN}
210
MN3
220
MN2


VDD
305
310
$4I_{BIAS}$
I_{BIAS}
I_{IN}
I_{OUT}
I_1
I_2
320
I_3
325
I_4
I_5
V_{OUT}
310
MN5
MN1
MN8
320
MN7
MN4
MN3
315
MN6
MN2


VDD
405
I_{IN}
I_{BIAS}
$4I_{BIAS}$
I_{OUT}
415
V_{OUT}
V_{CASC-N}
MN6
MN1
425
MN4
MN3
420
MN7
MN5
MN2


VDD
I_{IN}
I_{BIAS1}
I_{BIAS1}
I_{OUT}
515
V_{OUT}
MN4
505
510
MN1
MN6
S-F LEVEL SHIFTER
MN5
S-F LEVEL SHIFTER
520
MN3
MN2


VDD
MP3
MP1
555
560
S-F LEVEL SHIFTER
MP5
MP6
S-F LEVEL SHIFTER
MP4
MP2
I_{IN}
I_{BIAS1}
I_{BIAS1}
I_{OUT}

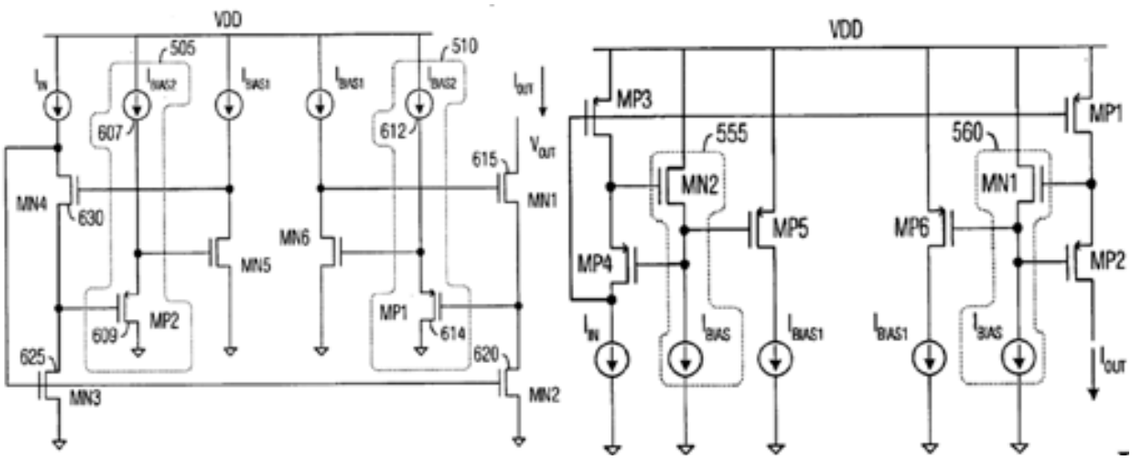


Fig. 12. Structure features for cascode current source with wide output swing circuits

3.1.4 Structure features for oscillators

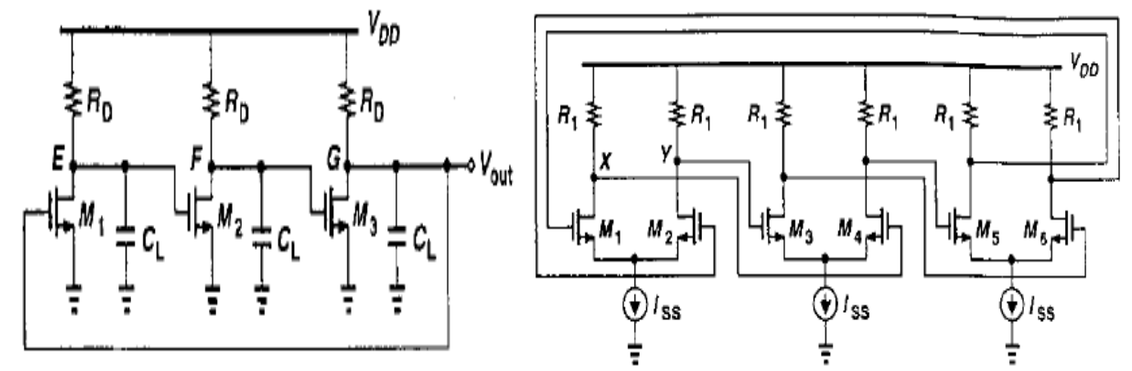
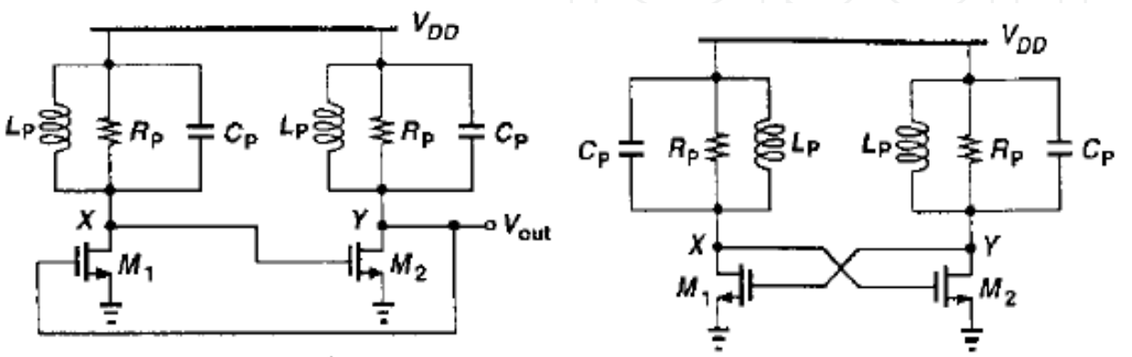


Fig. 13. Structure features for ring oscillators



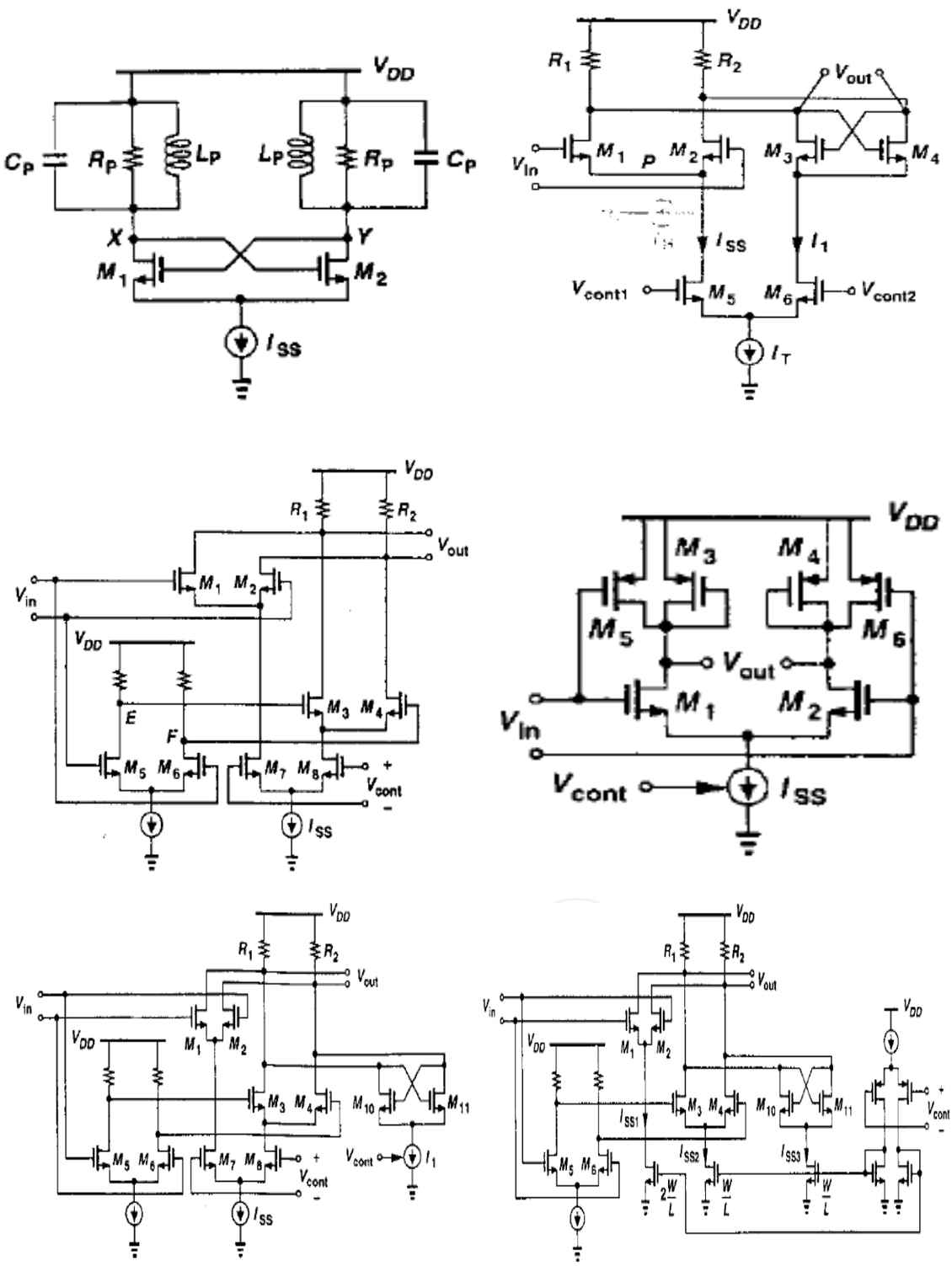


Fig. 14. Structure features for cascade oscillators

3.1.5 Structure features for charge pump

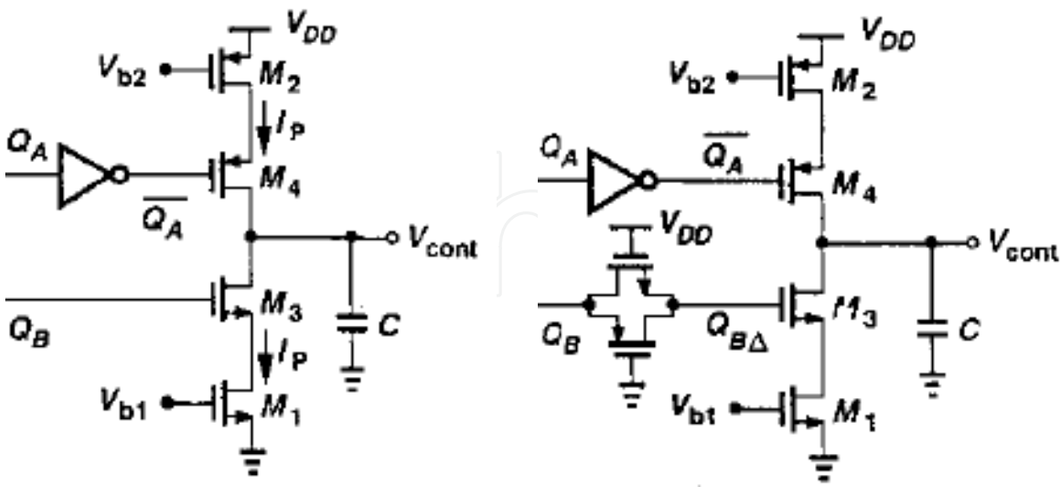
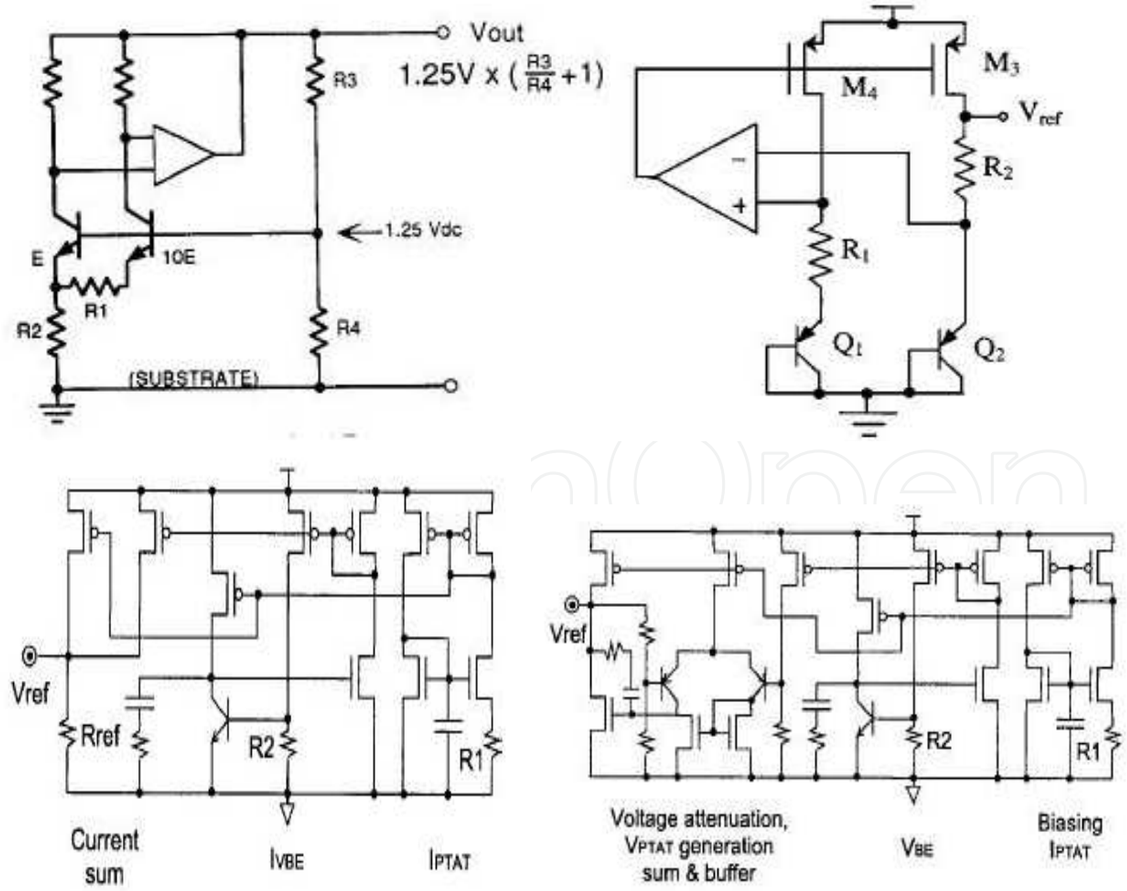


Fig. 15. Structure features for charge pump circuits

3.1.6 Structure features for band-gap circuits



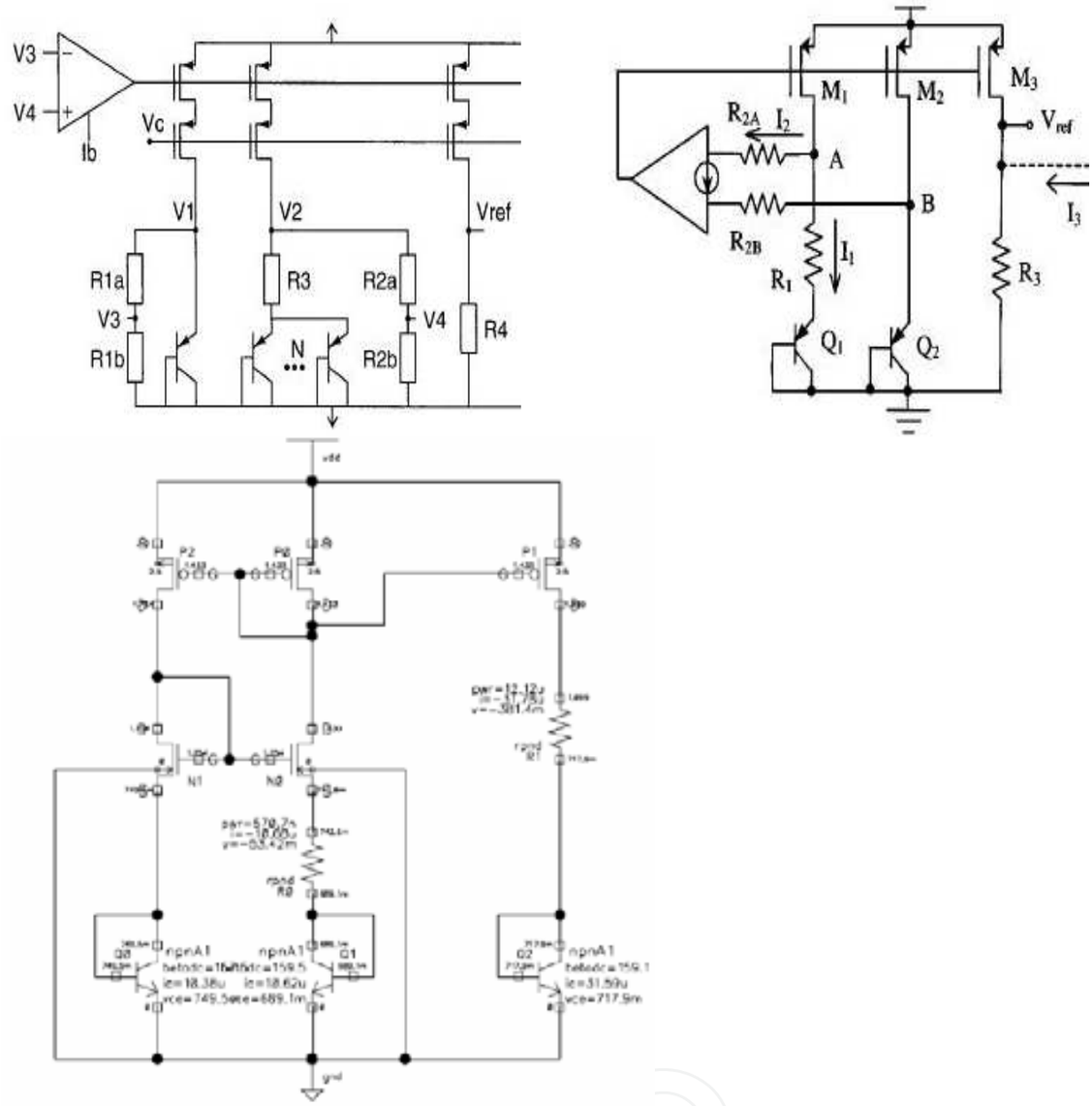


Fig. 16. Structure features for band gap circuits

3.2 High level analog structure features[1-3]

3.2.1 Structure features for OPA and OPA-based circuits

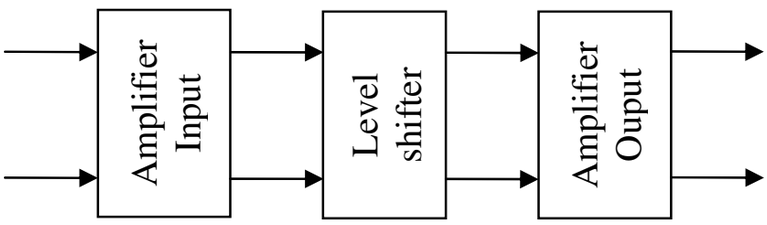


Fig. 17. Structure features for OPA circuits

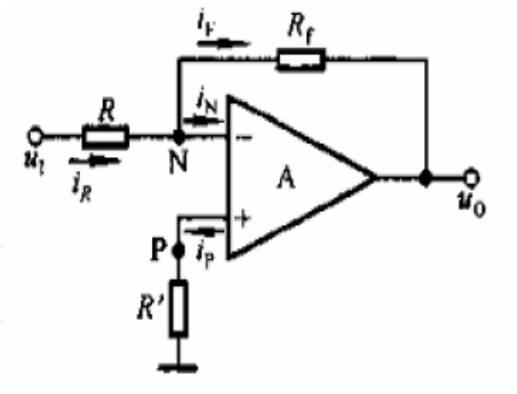


Fig. 18. Structure features for INV-Ratio circuit

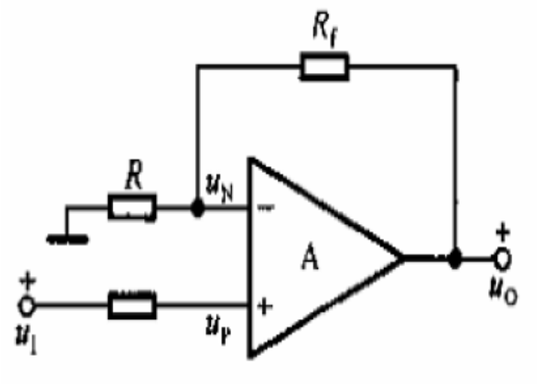


Fig. 19. Structure features for PASS-Ratio circuit

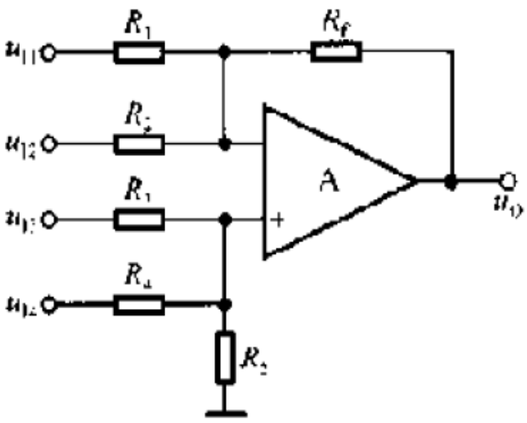


Fig. 20. Structure features for sum circuit

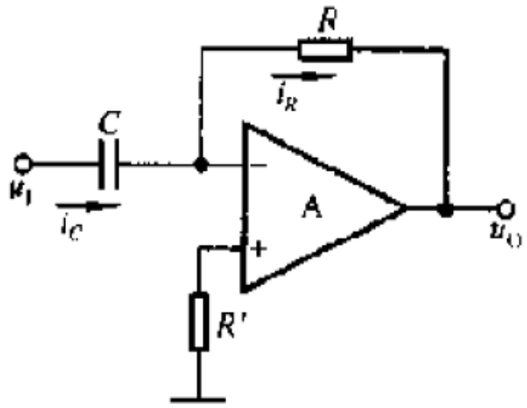


Fig. 21. Structure features for differentiator circuit

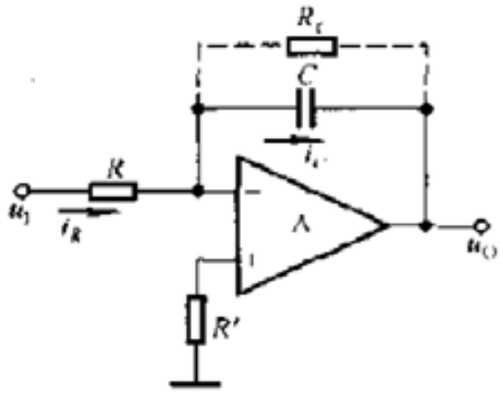


Fig. 22. Structure features for integrator circuit

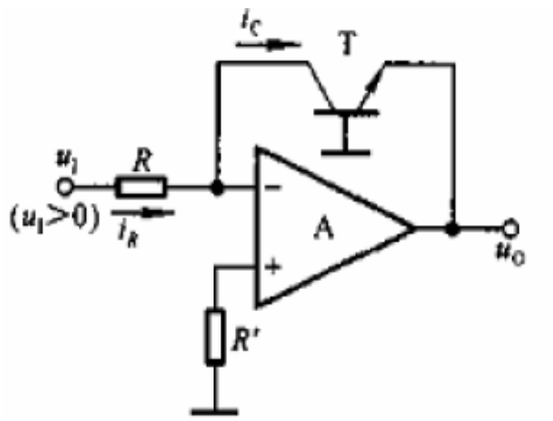


Fig. 23. Structure features for logarithm circuit

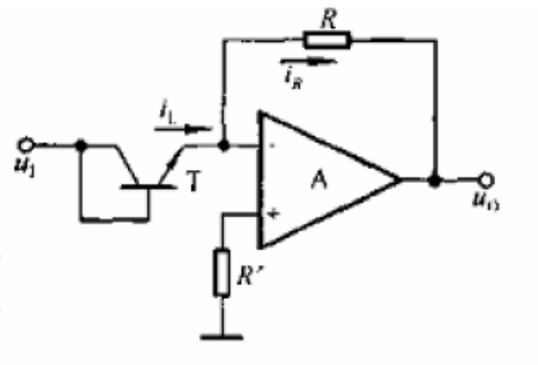


Fig. 24. Structure features for exponential circuit

3.2.2 Structure features for active filtering circuits

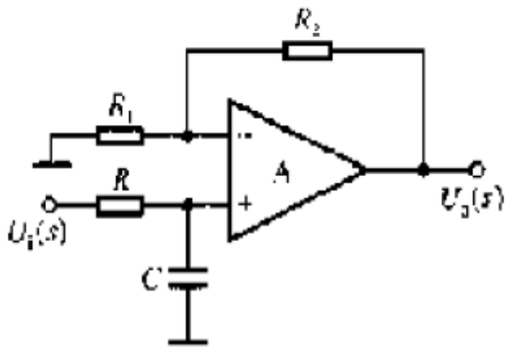


Fig. 25. Structure features for Low-pass (1st-order) filter circuit

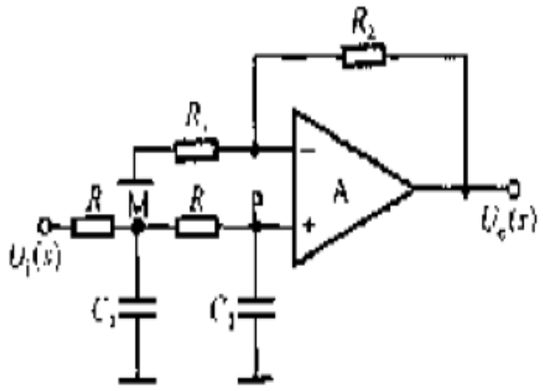


Fig. 26. Structure features for Low-pass (2nd order) filter circuit

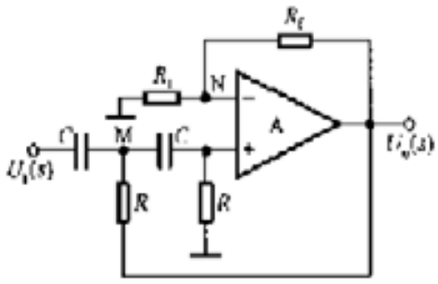


Fig. 27. Structure features for high-pass filter circuit

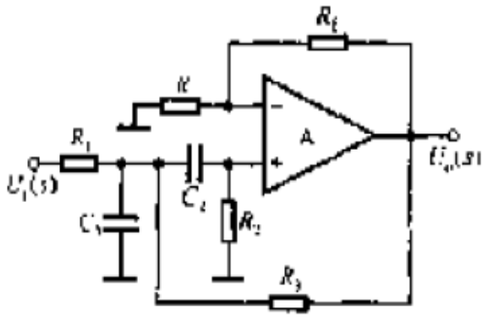


Fig. 28. Structure features for band-pass filter circuit

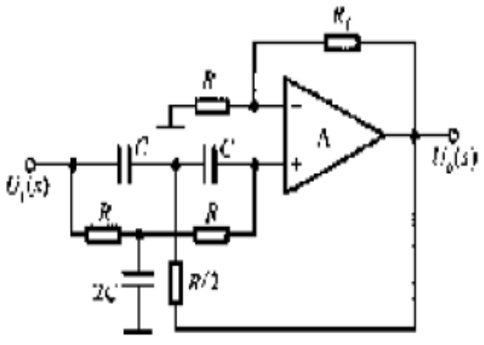


Fig. 29. Structure features for Band-resistive filter circuit

3.2.3 Structure features for signal transformation circuits

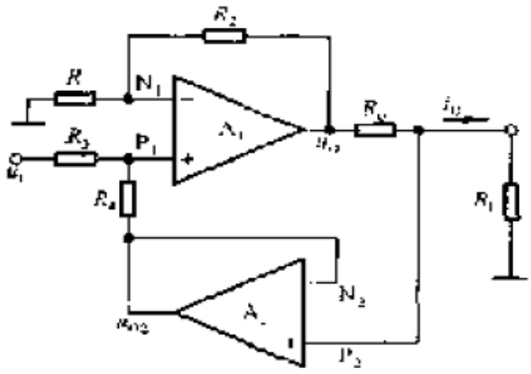


Fig. 30. Structure features for voltage / current transformation circuit

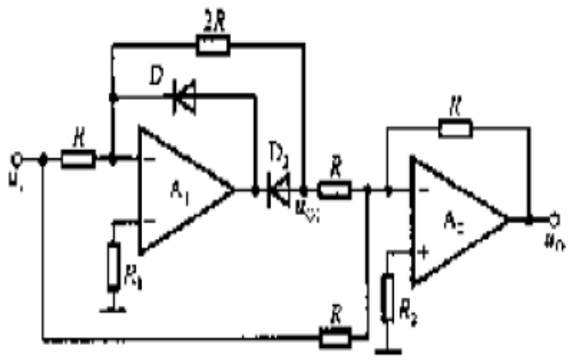


Fig. 31. Structure features for AC/DC transformation circuit

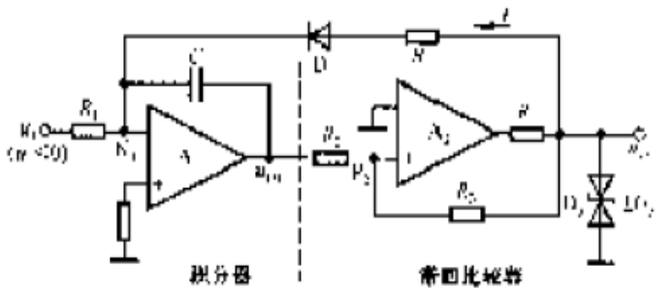


Fig. 32. Structure features for Voltage / frequency transformation circuit

3.2.4 Structure features for PLL

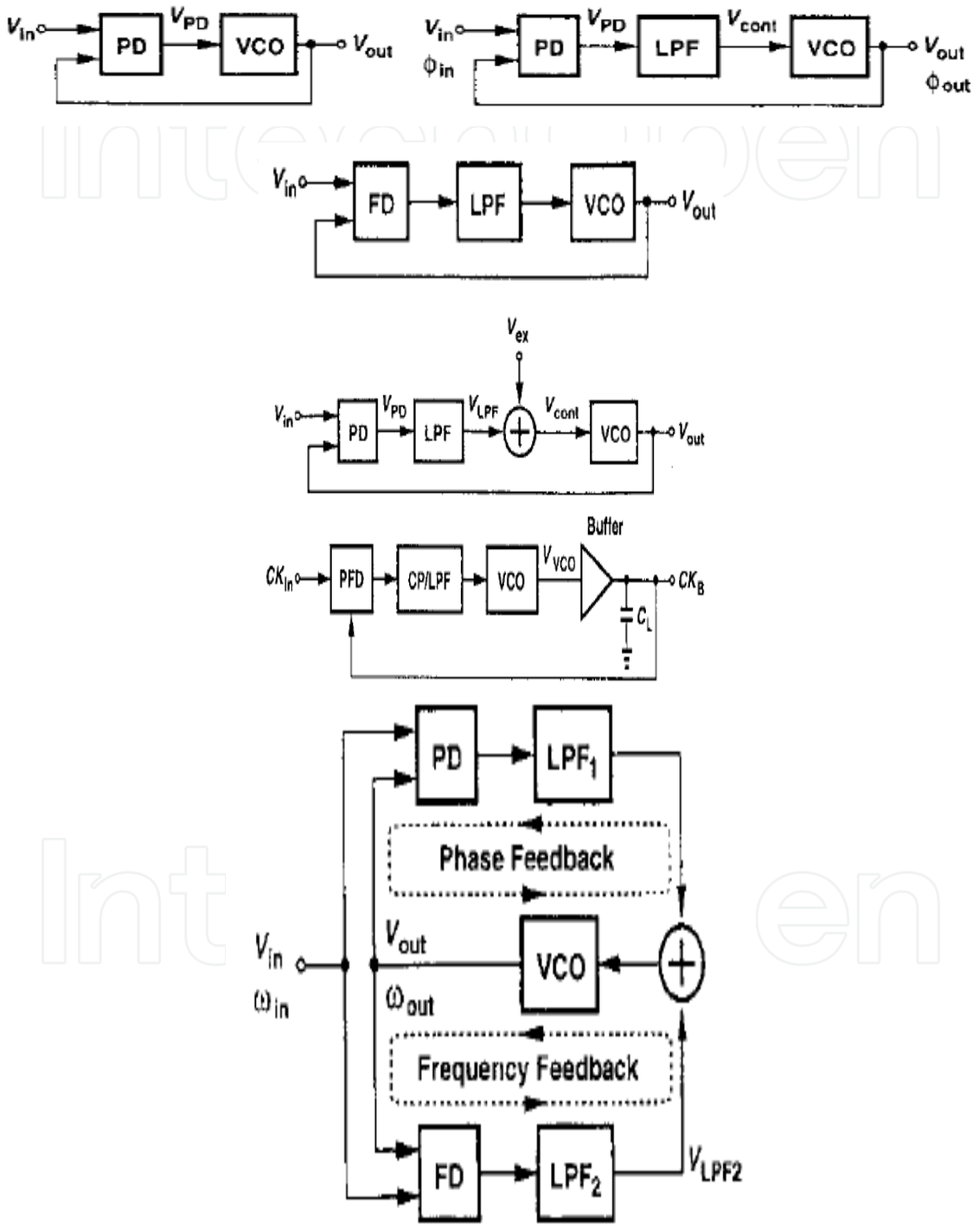


Fig. 33. Structure features for PLL circuits

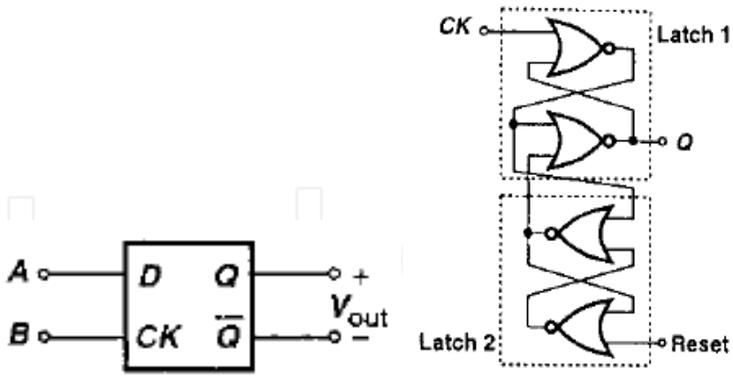


Fig. 34. Structure features for D-FF as PD

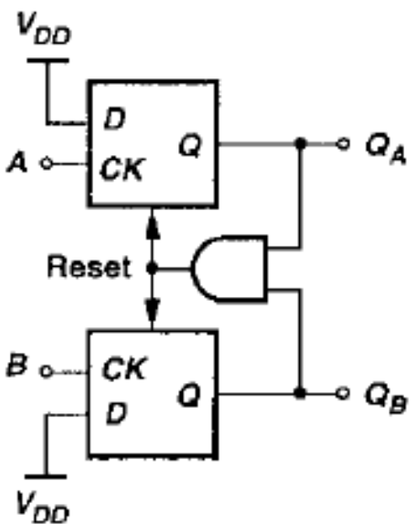


Fig. 35. Structure features for PFD circuit

3.2.5 Structure features for A/D Converters

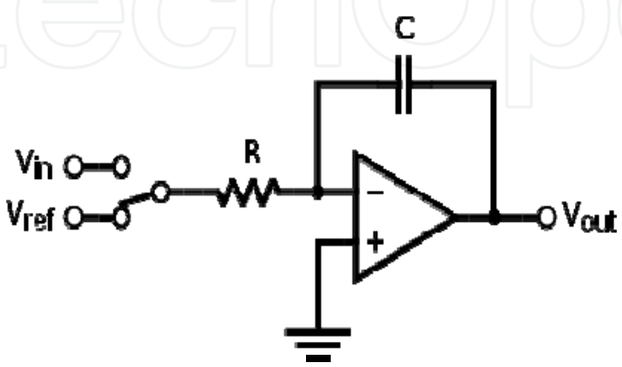


Fig. 36. Structure features for integrating ADC

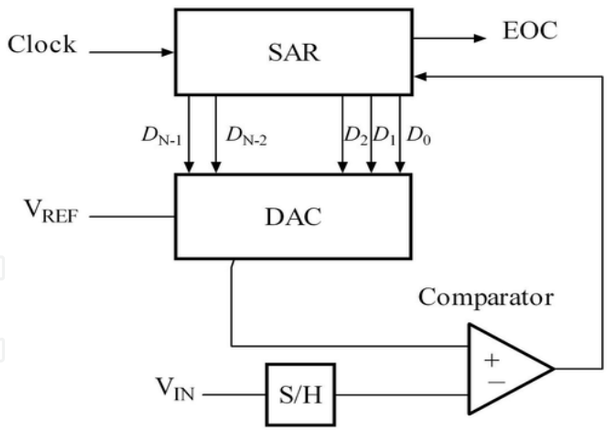


Fig. 37. Structure features for successive approximation ADC

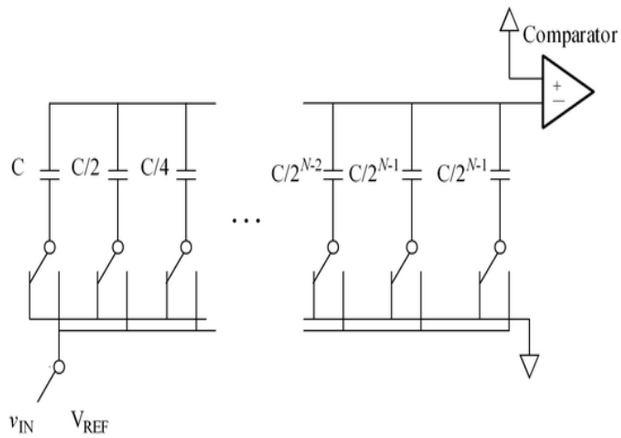


Fig. 38. Structure features for charge-redistribution SA-approximation ADC

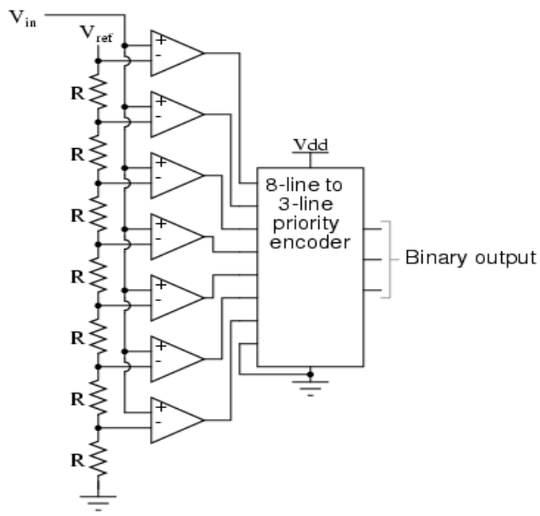


Fig. 39. Structure features for flash ADC

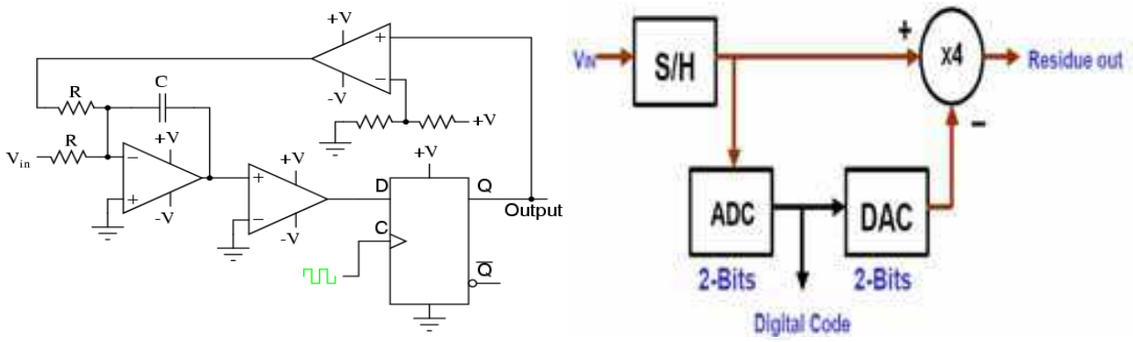


Fig. 40. Structure features for Σ-Δ ADC

3.2.6 Structure features for DAC

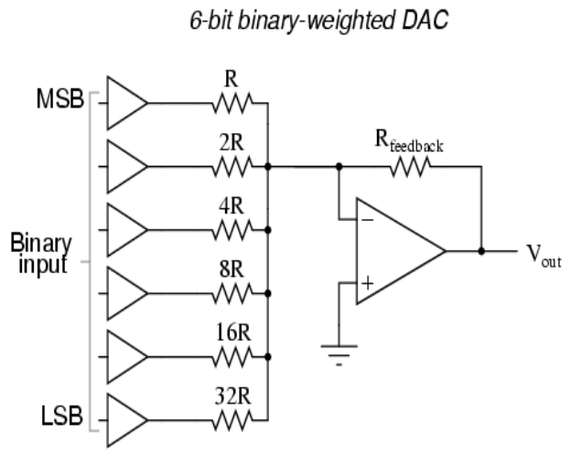


Fig. 41. Structure features for R/2nR DAC

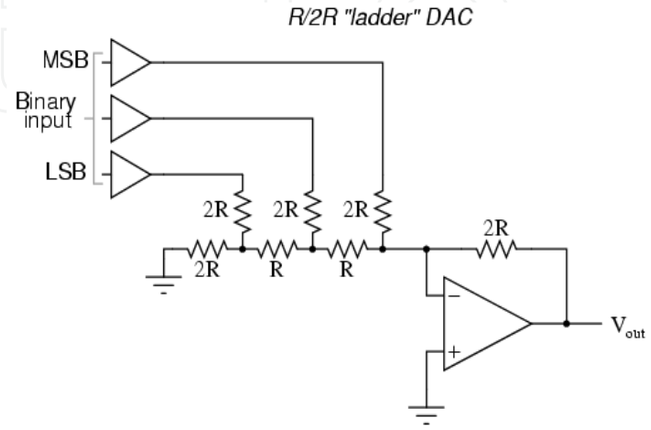


Fig. 42. Structure features for R/2R DAC

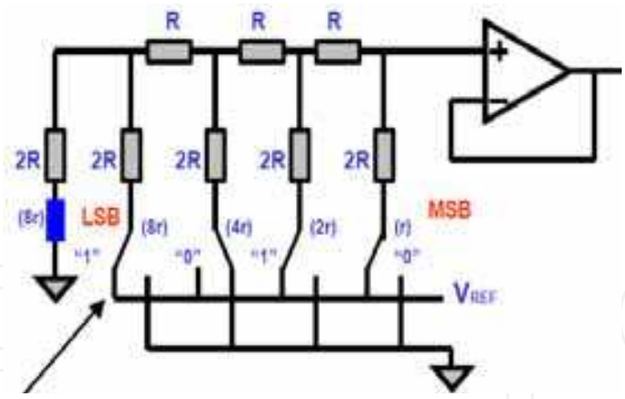


Fig. 43. Structure features for voltage scaling DAC

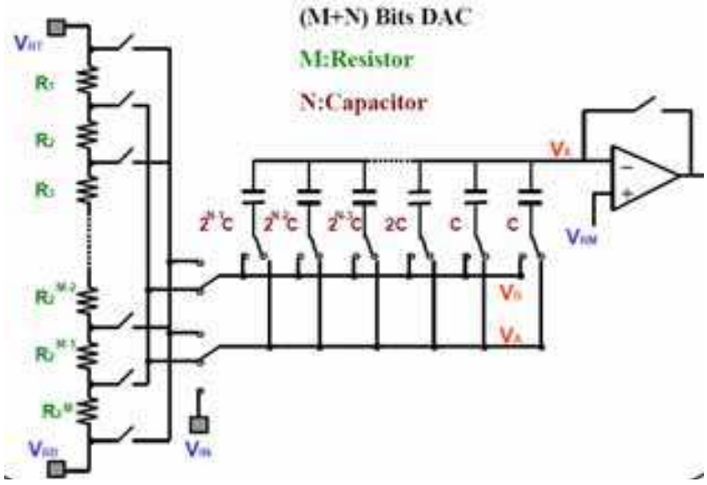


Fig. 44. Structure features for voltage and charge scaling DAC

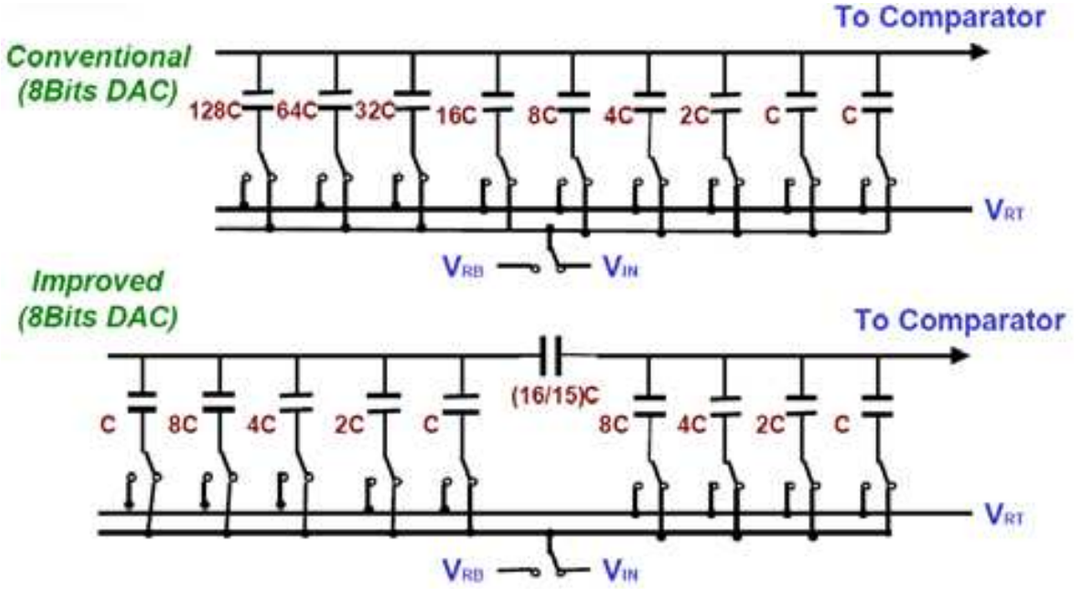


Fig. 45. Structure features for charge scaling DAC

3.3 Structure feature library composition

The structure feature library mainly contains structure feature description enclosed with a cell in SPICE netlist format, the cell name consists of keyword as prefix, the separator char "-", and a normal string for making cell name be unique, where the keyword represents the functionality of the analog structure.

For the bottom level analog structure feature description, device level netlist is used to describe the devices and their interconnections; and for the high level analog structure feature description, the block level netlist is used to describe the member block instantiations and their interconnections, the member block instantiation comes from a low level block of specific functionality, i.e., the template cell name quoted in the member block instantiation must be a keyword representing functionality rather than a specific cell name, which means that the instantiation represents the instantiation of functionality rather than the instantiation of a specific structure, which makes high level structure feature description independent from the specific detail low level or bottom level analog structure.

3.4 Structure feature associated attributes

Structure feature associated attributes include the constraints for schematic synthesis, sizing, floorplanning, layout, symbol shape, pin-out attributes, and others.

3.4.1 Schematic constraint knowledge

Constraints for schematic generation and optimization should include the constraints within a direct current path, the constraints between direct current paths, the constraints between blocks, and terminal placement constraints.

The constraints within a direct current path include the device list of direct current path, the top to down device sequence from power to ground based on power reaching level, and the device symmetry between direct current path branches.

The constraints between direct current paths include the device symmetry among the direct current paths, the parallel direct current paths of same signal reaching level, and the left to right direct current path sequence from input to output based on signal reaching level for direct current paths.

The constraints between blocks include the symmetry between the blocks, the left to right sequence from input to output based on signal reaching level for blocks, the ring sequence of the blocks based on signal path ring, and the parallel blocks based on signal reaching level.

The terminal placement constraints include the side constraint, the top to down sequence for left side and right side terminals, and the left to right sequence for top side and bottom side terminals.

3.4.2 Sizing constraint knowledge

Constraints for circuit design and optimization [11][13][22] can merge the optimization parameters, reduce the exploration space, and speed up the optimization for sizing procedure, so it is very important to generate such constraints no matter how the sizing step is implemented in hand or in automation.

Structure constraints for transistor pairs can be set up for differential pairs, level shifter, complementary pairs, current mirrors, matched direct current path, and matched blocks in future, so the first step for structural constraint generation is to execute the low level structure feature base matching exploration and high level structure feature based matching

exploration, which is described before, then set up such structure constraints for those device pairs with the following considerations.
For good mismatch properties and an area efficient layout, the channel lengths and the finger channel widths of the two transistors must be the same respectively. The ratio of the two transistor finger numbers must be equal to the ratio of the currents, although the ratio is *1* for differential pairs and current mirrors, and 1 or other integer values for others.

$$L_{M1} = L_{M2},\ FW_{M1} = FW_{M2},\ \text{and}\ I_1 / I_2 = FM_{M1} / FM_{M2}$$

The smaller the area of a transistor, the higher is its mismatch sensitivity. Therefore the transistor channel width and length must not fall below a minimum value W_{min} and L_{min} for differential pairs, level shifter, complementary pairs, current mirrors, and current sources:

$$FW_i * FM_i \geq W_{min} \ \text{ and } \ L_i \geq L_{min,} \quad i \in \{M1, M2, \ldots\}$$

Both transistors operate as voltage-controlled current sources (vccs) and thus they must be in saturation for current mirrors and current sources:

$$0 < V_{DSi} < V_{Gi} = V_{GSi} - V_T, \quad i \in \{M1, M2, \ldots\}$$

For a low VT-mismatch sensitivity, the effective gate voltage must not fall below a minimum value V_{Gmin} for current mirrors and current sources:

$$0 < V_{Gmin} < V_{GSi} - V_T, \qquad i \in \{M1, M2, \ldots\}$$

For a low λ sensitivity the difference of the drain source voltages must not exceed a maximum value V_{DSmax} for current mirrors and current sources:

$$| V_{DSM1} - V_{DSM2} | < V_{DSmax}$$

3.4.3 Layout constraint knowledge

Constraints for layout design and optimization [4-7][16-21][23-36] include the symmetry constraints for devices, direct current path branches, direct current paths, blocks and upper level circuits, the matching constraints for group of devices, the neighboring constraints, the protection constraints, the signal path and sequence constraints for direct current paths, and the direct current path and power reaching sequence constraints for group of devices.
The symmetry constraints can be used for minimizing the mismatch by mirroring placement of devices, direct current path branches, direct current paths, blocks, or upper level circuits, and mirroring the wiring of interconnections to reduce the mismatch on devices and the mismatch on wires, in further to reduce mismatch on direct current path branches, direct current paths, blocks and upper level circuits during layout design and optimization, and such constraints can be gotten with encoding based symmetry direction.
The matching constraints can be used for minimizing the mismatch on devices, direct current path branches, direct current paths, and upper level circuits by optimal placement of matching mode and dummy insertion to reduce the mismatch due to parasitic and process variations, such constraints can be gotten from structural feature based recognition for devices, encoding based match recognition for direct path braches, direct current paths, blocks, and upper level circuits.
The neighboring constraints can be used for minimizing the interconnection parasitic, interconnection interference, and interference among neighboring devices, which includes closing-necessary, neighboring-forbidden, and less than / far away from a specified distance.

The protection constraints can be used for preventing the critical devices or critical device groups interfered electrically by others, such constraints can be gotten from the previous signal path tracing and matching device exploration method.
The signal path and sequence constraints for direct current paths can be used for minimizing the interconnection parasitic on signal path to ensure the circuit frequency performance while layout design and optimization, and such constraints can be gotten from the signal path tracing method.
The direct current path and power reaching sequence constraints for group of devices can be used for minimizing the interconnection parasitic on direct current path so as to reduce the dc operation point variation due to parasitic on such path and ensure the DC performance while layout design and optimization, and such constraints can be gotten from the direct current path tracing method.

3.4.4 Constraint knowledge extraction based on good example circuits

Structure feature associated constraints are obvious in part, such as matching between differential pair devices and matching among current mirror / current source devices, but most of them are not so clear, so they need to be setup by hand based on the designer's professional experiences, it is very effective, but low efficiency due to handwork. There also exists another way to setup part of those constraints with the leverage of some good example circuits, which have embedded more professional design experiences.
Constraint knowledge extraction based on good example circuits mainly includes 1) analog structure feature analysis, 2) locating for analog structure feature devices / blocks, and 3) constraint capture for analog structure features from good schematic and layout data using geometry calculation, such as one level symmetry and multi-level symmetry, matching and matching mode, neighboring, protection, and so on.

3.5 Structure feature recognition

Recognition of low level analog structure feature is mainly graph-isomorphism of devices and connections, and recognition of high level analog structure is mainly graph-isomorphism of function blocks and interconnections with the ignorance of detail bottom devices and interconnections among them, it is to say that two high level blocks may have same functions if they have same composition of basic or high level functional blocks and interconnections although their corresponding low level functional blocks of the identical functionality may have different composition of devices and interconnection.

3.5.1 Recognition for low level analog structure features

Recognition for low level analog structure features is a direct searching procedure for complete matching on detail devices and connections among them between the source analog structure and the analog structure feature template with a bit tricky for speeding up.
As shown in Fig. 46, the main steps include graph setting-up, encoding for source analog structure, finding matched low level analog structure templates from template map using source structure coding value, and getting the functionality coding value for up level structure feature recognition and the associated attributes. The template map is setup from the analog structure feature template library.

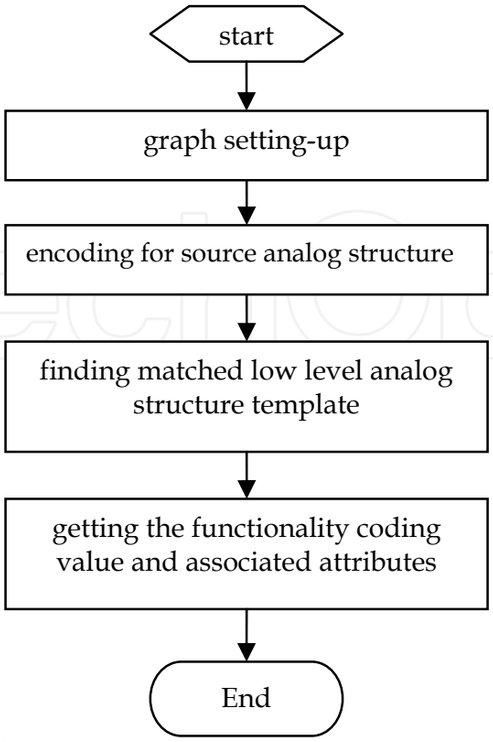


Fig. 46. Procedure for low level analog structure feature recognition

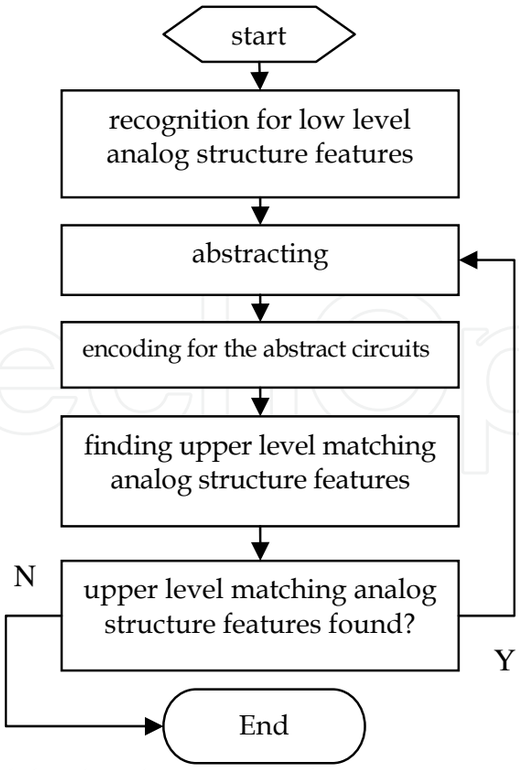


Fig. 47. Procedure for high level analog structure feature recognition

3.5.2 Recognition for high level analog structure features

Recognition for high level analog structure features is an iterative abstracting and searching procedure for complete matching on functional blocks and connections among them but with the ignorance of their bottom detail devices and connections between the source analog structure and analog structure feature template with a bit tricky for speeding up.

As shown in Fig. 47, the main steps include 1) recognition for low level analog structure features, 2) abstracting, i.e., replacing low level analog structure with virtual functional block with ignorance of detail composition, 3) encoding for the abstract circuits, and 4) finding the upper level matching templates with encoding value comparison, repeat step 2) to step 4) until no any upper level matching templates are found.

4. Analog circuit functionality analysis and partitioning

The proposed analog circuit functionality analysis and partitioning flow is shown as in Fig. 47. The input information includes the necessary information, such as circuit netlist and structural feature template libraries, and optional information: model type information and port information. The analysis and partitioning flow includes pre-processing netlist, tracing DC paths, tracing signal paths, encoding for DC paths and above block, checking isomorphism, and partitioning & res-constructing design in new hierarchy.

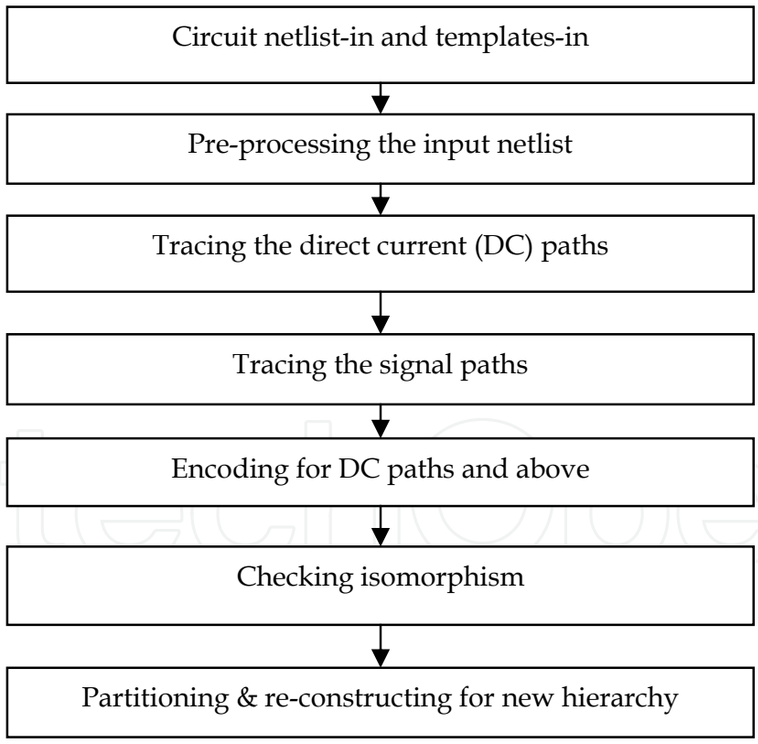


Fig. 48. Functionality analysis and partitioning flow

Analog functionality analysis is one of the bases for analog-aware circuit schematic synthesis; it is very different with traditional symbol analysis, it analyzes circuit functionality based on the functionality-known detail bottom level unit circuit templates, and the functionality-known complex high level circuit template with functionality

abstraction but without detail circuit descriptions for bottom unit circuits, which means that analog functionality analysis is an accurate pattern matching for low level unit circuits, and fuzzy pattern matching for high level circuits because the bottom devices and connections are ignored as possible and the bottom level unit circuits are represented by functionality and port connection only. The pattern matching is supported by encoding of graphic of devices, functional blocks, and connections among them and encoding value matching.
After functionality analysis, the analog design needs to be reconstructed with a new hierarchy based on functionality so as to use symbol templates to generate symbols and use the constraint templates to produce the accurate sizing, floor-planning, and layout constraints of the current analog circuit for future use. Also performance spec can be allocated into new hierarchy for future parallel on circuit optimization.

4.1 Input information

The input information for analog schematic synthesis includes the circuit netlist in spice netlist format, the data-in for mapping between devices & symbols, and the templates for analog structure features and associated templates as necessary inputs, and the partial port attributions or port name conventions as optional inputs.

4.2 Pre-processing

To make analog schematic synthesis more effectively, the pre-processing is necessary before core analog schematic synthesis procedure. The pre-processing includes identifying the aided devices, such as dummy devices and electronic static discharge (ESD) devices [45], removing them for analog structure feature analysis, port attribution passing, and internal power supply recognition.
The port attribution passing includes the top to down passing and the bottom up to top passing, which should be executed iteratively until all the port attributions are set for each cell especially when internal voltage regulation circuits are used for whole or part of the circuit, because the port attribution may be passed from one cell A to another cell B of same hierarchy level, for an example, cell A is a voltage regulator providing power supply to cell B.
Port attribution passing can set up the port attribution of each terminal for each cell, which can reduce the complexity of analog functionality analysis and other derived analysis, because the port attribution, such as power terminals, ground terminals, signal input terminals, and signal output terminals, can be used to limit the start points and the end points for current flow spreading and signal flow spreading, and the port attribution, such as power terminals and ground terminals can be used reduce the complexity of circuit-based graph especially.
To make port attribution passed smoothly, the internal power supply recognition is a necessary to make the internal power supply be regarded as power terminals of other internal circuits when the internal voltage regulation circuits are used so as to ease the analysis of other internal circuits. The internal power supply recognition should include band-gap structure feature recognition, band gap reference circuit identification by finding the OPA associated with the band-gap feature, and determination of output terminal(s) of the band gap reference circuits.

4.3 Tracing direct current paths

In the method operation of tracing the direct current paths, tracing can be spread along the direct currently flow direction, as shown in Fig. 49, or along the inverse of direction, which

is determined according to the presented terminal types, such as the positive power supply terminals, the ground terminals, the negative power supply terminals, the current mode input terminals, and the current mode output terminals. The detail tracing can be done as the following descriptions.

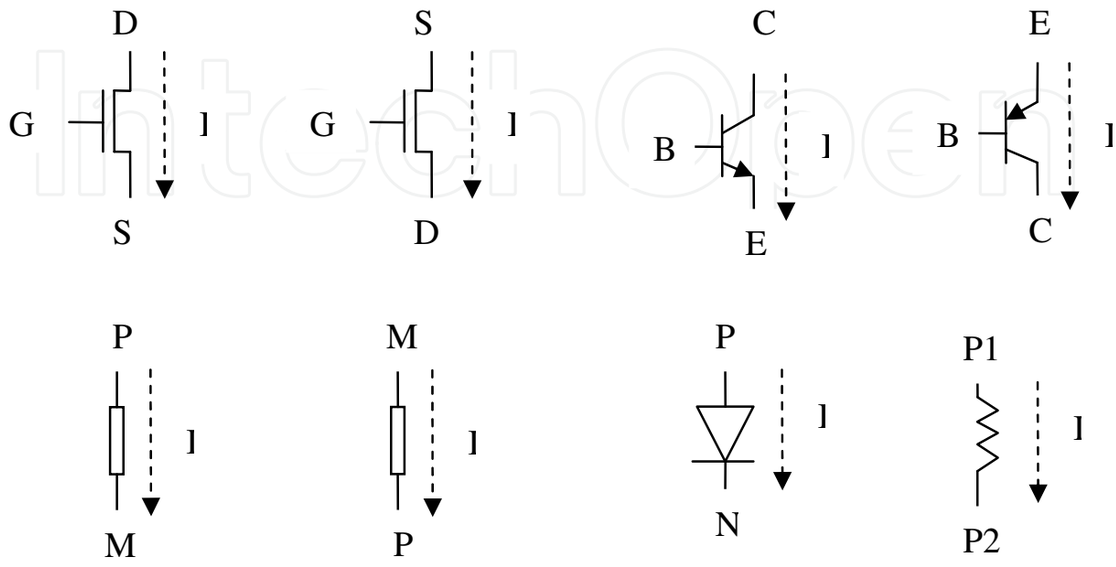


Fig. 49. Direction of current flow through devices

As the first operation method, the direct current path tracing can start from the positive power supply terminals or current mode input terminals; spread along the drain to source or source to drain for MOSFET and JFET, the collector to emitter for NPN BJT devices, the emitter to collector for PNP BJT, the positive terminal to negative terminal for diode, and one terminal to another terminal for some resistors and inductors, as shown in Fig. 50 and stop while reaching the ground terminals, negative power supply terminals, current output mode input terminals, or current mode output terminals. From such traversing, it gets the list of devices of a direct current path, calculates the minimum distance to the positive power supply terminals or current mode input terminals for each device, then sorts the device based on the distance values from min to max to get the device sequence of the current path.

As the second operation method, the direct current path tracing can start from the ground terminals, spread as above description, as shown in Fig. 51, and stop while reaching the negative power supply terminals. From such traversing, it gets the list of devices of a direct current path, calculate the minimum distance to the ground terminal for each device, then sort the device based on the distance values from min to max to get the device sequence of the current path.

As the third operation method, the direct current path tracing can start from the ground terminals, spread as the inverse of current flow direction, as shown in Fig. 52, and stop while reaching the current mode input terminals or the current mode output terminals. From such traversing, it gets the list of devices of a direct current path, calculate the minimum distance to the ground terminal for each device, then sort the device based on the distance values from max to min to get the device sequence of the current path.

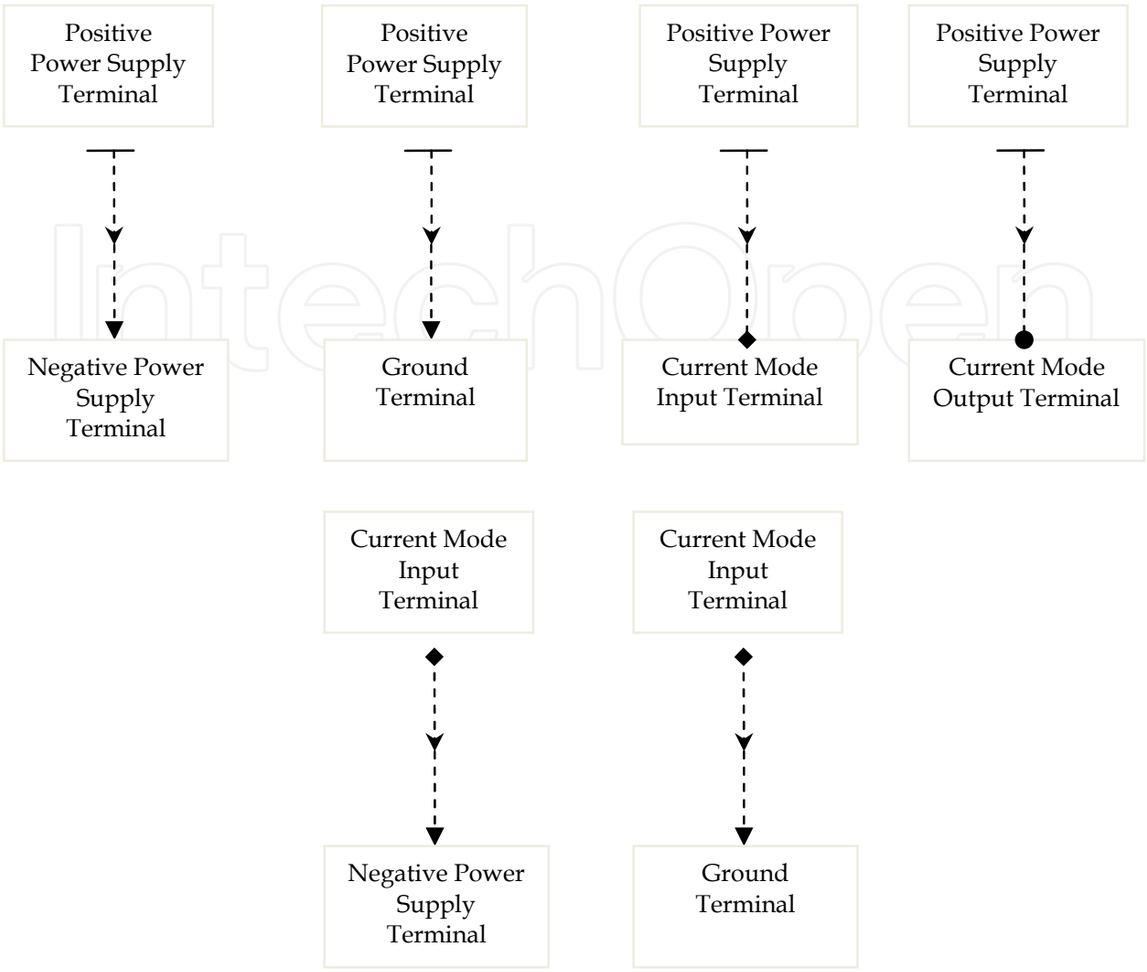


Fig. 50. Find the direct current path from the positive power supply terminal to the negative power supply terminal, the ground terminal, the current mode input terminal, and or the current mode output terminal, and from the current mode input terminal to the negative power supply terminal or the ground terminal with normal direct current direction

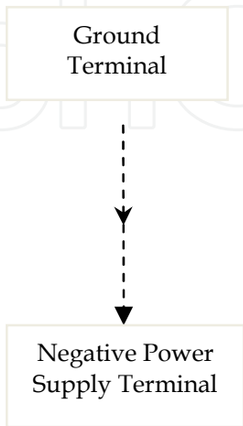


Fig. 51. Find direct current path from the ground terminal to the negative power supply terminal with the normal direct current direction

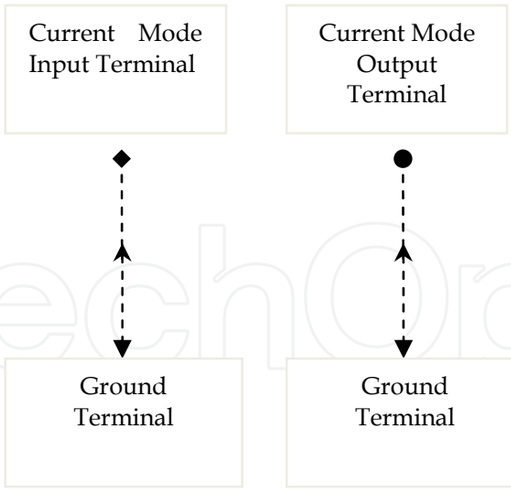


Fig. 52. Find the direct current path from the ground terminal to the current mode input terminal and from the ground terminal to current mode output terminal with reverse of direct current direction

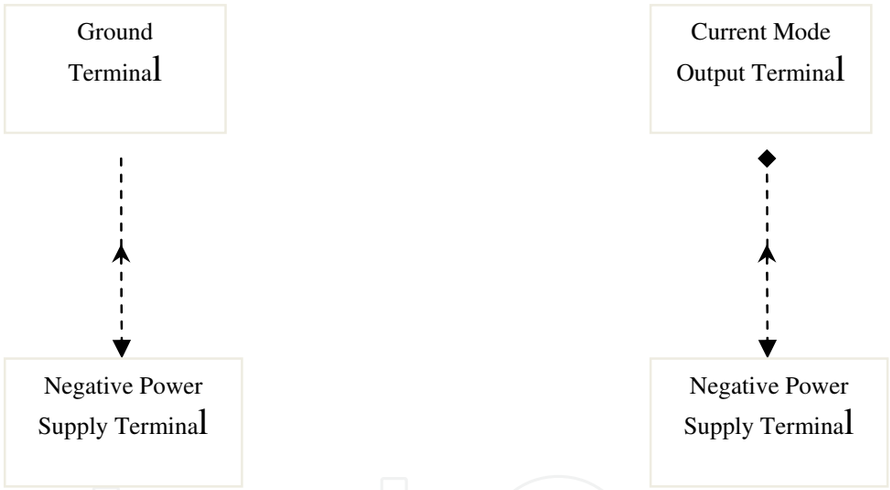


Fig. 53. Find the direct current path from the negative power supply terminal to the ground terminal or the current mode output terminal with reverse of direct current direction.

As the fourth operation method, the direct current path tracing can start from the negative power supply terminals, spread as the inverse of current direction, as shown in Fig. 53, and stop while reaching the ground terminals or current mode output terminals. From such traversing, it gets the list of devices of a direct current path, calculate the minimum distance to the negative power supply terminal for each device, then sort the device based on the distance values from max to min to get the device sequence of the current path.

For a typical circuit, any one of the above operation method cannot dig out all the direct current paths, so in practice, the combination of them is used, although there are some overlaps among the above four operation methods. To filter out the overlapping direct current path result, a map for identifying the handled devices is used so as to avoid unnecessary repeat operations.

As an addition, grouping devices of the current source are not in the same direct current path, but they are searched out, such as the companion devices from different direct current paths of current sources circuit; also the other devices from different current paths but with same power reaching levels or same ground reaching levels are searched out, so that such devices can be placed on one horizontal line for easy wiring in schematic view.

4.4 Tracing signal paths

In the method operation of tracing the signal paths [14], tracing starts from the input signal terminals, and spreads along gate to drain/source or drain/source to source/drain for MOSFET and JFET, base to collector/emitter or collector/emitter to emitter/collector for BJT, the positive terminal to negative terminal for diode, and one terminal to another terminal for some resistors/capacitors/inductors, as shown in Fig. 54 other than feedback or bypassing filtering devices. The signal spreading is terminated while reaching power supply terminals, ground terminals, or output terminals.

During signal path tracing, the signal input terminal node is put into the signal node list, handle the devices connected to the signal node, spread the signal based on the above signal flow direction rules so as to find next possible signal nodes to which these devices are connected to, put the new signal nodes into the signal node list, and traverse the signal node list until all the signal nodes are handled. To speed up tracing signal path, a device map and a node map should be used for a circuit. A flag is marked for a device in the device map while a signal spreading is handled on that device in case of repeating signal spreading on the same device in the future. Also, a flag is marked for a node in the node map while a signal spreading is handled on that node in case of repeating signal spreading on the same node in the future.

The distance between an input signal port and a device is defined as the signal reaching level of that device under that signal; the signal reaching level of a device may consist of signal reaching minimum level and signal reaching maximum level, which reflects different signal flow paths to that devices.

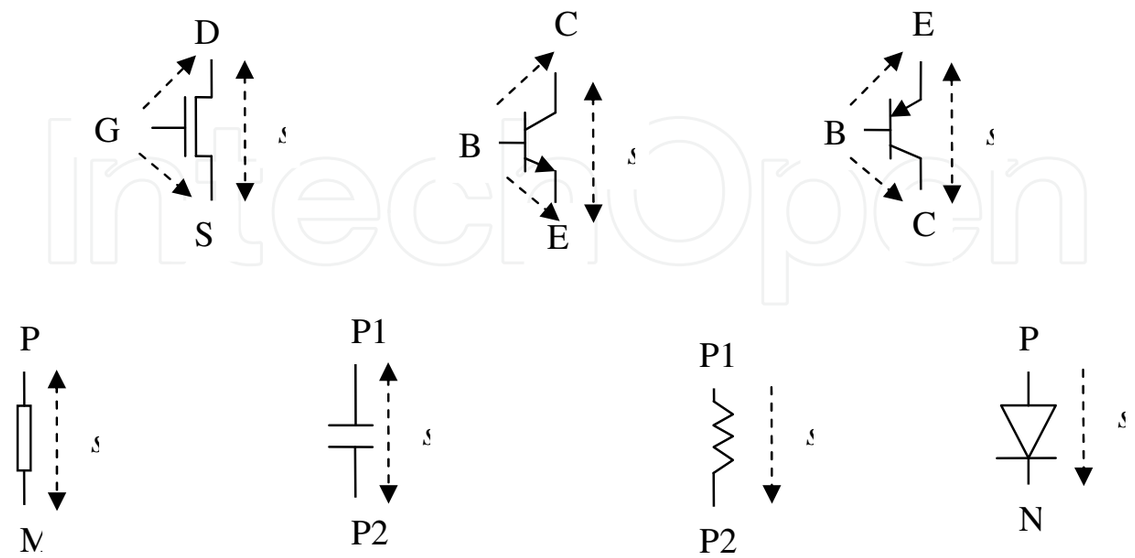


Fig. 54. Direction of signal flow through devices

Also, signal reaching level for a direct current path consists of the signal reaching minimum level and the signal reaching maximum level, they can be gotten from the minimum of signal reaching minimum levels and maximum of signal reaching maximum levels of all devices in such direct current path respectively.
In further, signal reaching level for a block consists of the signal reaching minimum level and the signal reaching maximum level, they can be gotten from the minimum of signal reaching minimum levels and the maximum of signal reaching maximum levels of all the direct current paths in such block respectively.

4.5 Encoding for blocks

Encoding from bottom level to up level, the bottom level is for direct current (DC) path only, and the up level is the combination of direct current paths and more.
To encode for a direct current path, try to find the matched DC path structural feature from the template libraries with ignorance of some auxiliary devices including the dummy devices, protection devices, MOSCAP devices, power-down devices, and biasing devices, assign the functionality name and functionality identification number to that DC path so that it is encoded with such identification number in a bit fuzzy logic.
To encode for a cell/block, each DC path is considered as a virtual block of a specific functionality, each sub-cell/block in the current cell domain is also considered as a black box of a specific functionality, so the encoding step is to try to find the matched template of same functional blocks and same signal connectivity among blocks, which can be handled as pattern matching issue on quasi one-dimension, the functionality name and functionality identification number is assigned to the current cell.

4.6 Checking isomorphism and quasi-isomorphism

In contrast to traditional sub-graph isomorphism algorithm [46-48], the checking issue is a quasi one-dimension graph due to the simplification from each DC path or clusters of DC paths to a functionality vertex, and also some unimportant connectivity is ignored, so it is a bit fuzzy logic. The computing complexity is closing to $O(n)$ due to the one dimension approximation and sequenced, so the encoding and code value comparison can be used efficiently for isomorphism checking.

4.7 Partitioning into hierarchy

The source circuit is abstracted in several hierarchy levels after the recognition of low level analog structure features and the recognition of high level analog structure features, and each block of any level in the abstract tree represents has a specific functionality. Reconstruct the circuit netlist based on such functionality recognition abstract tree, which includes the following main steps: 1) Determine the out connections of a block to build the port terminal information for that functional block 2) build the netlist for the functional block based on direct sub-blocks and their interconnection with sub-block handled as an instantiation of the corresponding sub-cell, and 3) build the netlist for the bottom level block: based on the detail devices and their interconnections. After that, such circuit partitioning can make the new hierarchical circuit more intuitive for designer to understand it and get more advantages on later circuit sizing, floorplanning and layout automation.

5. Constraint generation

Constraint generation is a very important step in analog schematic synthesis procedure [10]. After analog structure feature recognition, the analog structure feature associated constraint templates can be used to generate the constraints for schematic synthesis, circuit synthesis, and layout synthesis if the associated constraint template exists. The key is to find the device-to-device mapping relation and block-to-block mapping relation so as to replace the virtual device name or virtual block name with practical device name or practical block name of source circuits, it is very easy, herein we do not discuss about it. Here we focus on the case without constraint templates, as a complementary, the constraints can be generated with leverage of part of the analog structure feature recognition result and further analysis results.

5.1 Constraint generation for schematic generation and optimization

Constraints for schematic generation and optimization should include the constraints within direct current path, the constraints between direct current paths, the constraints between blocks, and the terminal placement constraints.

The constraints within a direct current path include the device list of direct current path, the top to down device sequence from power to ground based on power reaching level, and the device symmetry between direct current path branches. The first three constraints can be gotten as the result of tracing the direct current paths, and the constraint of device symmetry between direct current path branches can be checked out with the devices of the same power reaching level as a symmetry pair.

The constraints between direct current paths include the device symmetry among the direct current paths, the parallel direct current paths of same signal reaching level, and the left to right direct current path sequence from input to output based on signal reaching level for direct current paths. The first constraint can be checked out using sub-graph isomorphism method, the head line of the method can be overviewed as: 1) setup graph for each direct current path; 2) encode for each graph; 3) compare the encoding values; 4) if the encode values are matching, put the two direct current pats as symmetry candidate; and 5) check the signal reaching minimum level and signal reaching maximum level of the direct current paths of the candidate; regard them as symmetry pair if matching occurs. The second constraint can be checked out if any two direct current paths have identical the signal reaching minimum level and signal reaching maximum level. The third constraint can be checked out using the sorting based on the signal reaching minimum level and signal reaching maximum level.

The constraints between blocks include the symmetry between the blocks, the left to right sequence from input to output based on signal reaching level for blocks, the ring sequence of the blocks based on signal path ring, and the parallel blocks based on signal reaching level. The first constraint can be checked out if the two blocks are matched completely and have identical the signal reaching minimum level and signal reaching maximum level. The second constraint can be checked out by sorting the blocks with their signal reaching minimum levels and signal reaching maximum levels. The third constraint can be checked out by signal flow spreading, if a signal flow circle is checked, i.e., signal flow spreading meets a past checked signal points, all blocks on such signal flow circle construct the ring, the ring sequence of blocks are gotten by sorting with the signal reaching minimum levels and signal reaching maximum levels of those blocks. The fourth constraints can be checked out if any two blocks of a circuit have the identical signal reaching minimum levels and signal reaching maximum levels.

The terminal placement constraints include the side constraint, the top to down sequence for left side and right side terminals, and the left to right sequence for top side and bottom side terminals. For the side constraints, in principle, the input terminals are presented with left side constraint, the output terminals are presented with right side constraints, the positive power supply terminals are presented with the top side constraints, and the ground terminals and the negative terminals are presented with the bottom side constraints.

5.2 Constraint generation for circuit design and optimization

Constraints for circuit design and optimization can merge optimization parameters, reduce the exploration space, and speed up the optimization for sizing procedure, so it is very important to generate such constraints no matter how the sizing step is implemented in hand or in automation.

Structure constraints for transistor pairs can be set up for differential pairs, level shifter, complementary pairs, current mirrors, matched direct current path, and matched blocks in future, so the first step for structural constraint generation is to execute the low level structure feature base matching exploration and high level structure feature based matching exploration, which is described before, then set up such structure constraints for those device pairs with the following considerations.

For good mismatch properties and an area efficient layout, the channel lengths and the finger channel widths of the two transistors must be the same respectively. The ratio of the two transistor finger numbers must be equal to the ratio of the currents, although the ratio is *1* for differential pairs and current mirrors, and 1 or other integer values for others.

$$L_{M1} = L_{M2},\ FW_{M1} = FW_{M2},\ \text{and}\ I_1 / I_2 = FM_{M1} / FM_{M2}$$

The smaller the area of a transistor, the higher is its mismatch sensitivity. Therefore the transistor channel width and length must not fall below a minimum value W_{min} and L_{min} for differential pairs, level shifter, complementary pairs, current mirrors, and current sources:

$$FW_i * FM_i \geq W_{min} \ \text{and} \ L_i \geq L_{min,} \quad i \in \{M1, M2, \ldots\}$$

Both transistors operate as voltage-controlled current sources (vccs) and thus they must be in saturation for current mirrors and current sources:

$$0 < V_{DSi} < V_{Gi} = V_{GSi} - V_T, \quad i \in \{M1, M2, \ldots\}$$

For a low VT-mismatch sensitivity, the effective gate voltage must not fall below a minimum value V_{Gmin} for current mirrors and current sources:

$$0 < V_{Gmin} < V_{GSi} - V_T, \qquad i \in \{M1, M2, \ldots\}$$

For a low λ sensitivity the difference of the drain source voltages must not exceed a maximum value V_{DSmax} for current mirrors and current sources:

$$| V_{DSM1} - V_{DSM2} | < V_{DSmax}$$

5.3 Constraint generation for layout design and optimization

Constraints for layout design and optimization include the symmetry constraints for devices, direct current path branches, direct current paths, blocks and upper level circuits,

the matching constraints for group of devices, the neighboring constraints, the protection constraints, the signal path and sequence constraints for direct current paths, and the direct current path and power reaching sequence constraints for group of devices.
The symmetry constraints can be used for minimizing the mismatch by mirroring placement of devices, direct current path branches, direct current paths, blocks, or upper level circuits, and mirroring the wiring of interconnections to reduce the mismatch on devices and the mismatch on wires, in further to reduce mismatch on direct current path branches, direct current paths, blocks and upper level circuits during layout design and optimization, and such constraints can be gotten with encoding based symmetry direction.
The matching constraints can be used for minimizing the mismatch on devices, direct current path branches, direct current paths, and upper level circuits by optimal placement of matching mode and dummy insertion to reduce the mismatch due to parasitic and process variations, such constraints can be gotten from structural feature based recognition for devices, encoding based match recognition for direct path braches, direct current paths, blocks, and upper level circuits.
The neighboring constraints can be used for minimizing the interconnection parasitic and interconnection interference.
The protection constraints can be used for preventing the critical devices or critical device groups interfered electrically by others, such constraints can be gotten from the previous signal path tracing and matching device exploration method.
The signal path and sequence constraints for direct current paths can be used for minimizing the interconnection parasitic on signal path to ensure the circuit frequency performance while layout design and optimization, and such constraints can be gotten from the signal path tracing method.
The direct current path and power reaching sequence constraints for group of devices can be used for minimizing the interconnection parasitic on direct current path so as to reduce the dc operation point variation due to parasitic on such path and ensure the DC performance while layout design and optimization, and such constraints can be gotten from the direct current path tracing method.

6. Analog schematic generation

6.1 Symbol generation based on functionality

Generating the cell/block symbol based on its functionality includes the following sub-operations: determining the symbol pattern from a symbol shape template based on the functionality of the cell/block; determining the port terminal pattern for each port terminal symbol based on its port type; determining the side location for each port terminal symbol based on its port type; determining the sequence of the ports on each side based on the port terminal attribute; and determining the exact location for each port terminal pattern.

6.2 Symbol placement based on functionality

Determining the placement of the symbols of the devices, the ports, and the cells/blocks includes the following sub-operations: determining the placement of device symbols for the devices in the DC path; binding the placement of device symbols for the devices in the DC path as virtual block; determining the placement of the virtual blocks for the DC paths; tuning the placement for the device symbols; and placing the port terminal symbols.

In the operation of determining the placement of device symbols for the devices in the DC path, the symbols in a direct current path must be placed from up to down associated with the current flow direction (POWER to GROUND), which is identified with the direct current path analysis, the associated dummy devices and protection devices are also placed closing to the corresponded device symbols, and also symmetry requirement in a DC path is followed in this operation.
In the operation of binding the placement of device symbols for the devices in the DC path as virtual block, a DC path (including the associated dummy devices and protection devices) is regarded as a virtual block, and a rectangle is used as its symbol.
In the operation of determining the placement of the virtual blocks for the DC paths, the virtual block symbol placement is based on the signal reaching level which is determined by signal reaching level analysis step, and the virtual block is placed with signal reaching level incremental order from left to right. DC path symmetry requirements are also followed by specifying the symmetry axis and put make the virtual blocks of the symmetry pair mirrored with it to each other.
In the operation of tuning the placement for the device symbols, fine tuning includes: tuning the powered devices on the same top horizontal grid line; tuning the grounded devices on the same bottom horizontal grid line; tuning the MOSCAP devices direction for bridging source net and POWER/GROUND net; tuning the matching device symbols from the current mirror/source pair to make all the associated gate terminals on the same horizontal grid line; mirroring the diode-connected device symbol of current mirror/source; and tuning the rotation status and location of the symbol for the devices(no DC current) bridging between DC paths for shortest wiring length.
The block symbols in a cell are placed with the signal path folding minimized and the total wiring length minimized, and also parallel stages must be followed.
In the operation of placing the port terminal symbols, the port terminal placement is executed as: determining the side location for each port terminals based on the port type with IN on left side, OUT on right side, VCC on top side, and GND and VSS on down side; determining the port order(top to down on left and right sides, left to right for top and down sides for each side); binding the differential nets and bus nets in neighboring sequence; and tuning the exact location for wiring length minimized.

6.3 Wiring based on functionality

In the operation of wiring for schematic, the wiring includes the special wiring, wiring in a direct current path, wiring between neighboring direct current paths, wiring among multiple direct current paths, and wiring among cell instances and blocks.
Special wirings include the wiring for the net among differential devices and tail current devices, the wiring for differential net pair, the wiring for the bus/bundle nets, the wiring among current mirror and current source devices, the wiring for dummy devices, the wiring for MOSCAP devices, the wiring for cross link between two DC paths, the wiring for dummy devices, the wiring for the protection devices, the wiring for the bridging devices, and the wiring for POWER and GROUND nets.
Wiring in a DC path includes the major vertical wiring with high weight and the minimum horizontal wiring with low weight as transition only.
Wiring between the neighboring DC paths includes the major horizontal wiring with high weight and the minimum vertical wiring with low weight for transition only.

Wiring among DC paths is similar with the wiring between neighboring DC paths, the most difference is that the wiring needs to take the wiring overlapping the device symbol and other wiring cross-points into account. For this reason, a line exploration algorithm can be used with device symbol and other wiring cross-points handled as the obstacles with safety halos.
Wiring among cell / block instances is similar with the wiring among DC paths, the most difference is that both horizontal and vertical wiring have the same possible occurrence, so they have the same weights in the cost of wiring.

6.4 Constraint identification

After the placement and the wiring processes, the identification on the schematic is necessary to make the circuit engineer and the layout engineer have a good insight on the design for circuit optimization, floorplanning and layout optimization. The identification includes the identification of the symmetry requirements in a DC path, the identification of the device matching requirements among DC paths, the identification of the symmetry requirements between DC paths, the identification of the dummy devices, the identification of the protection devices and the associated protected devices, the identification of the MOSCAP devices, the identification of the critical signal nets, and the identification of the net current and net wiring width.
All the identification contents are generated by structural feature based circuit functionality analysis and partitioning engine.

7. Analog-aware schematic synthesis with companion circuits

The professional designers have a very good thumb of rules on drawing analog circuit schematic, and it is necessary to follow such rules to make circuit schematic more analog-aware while drawing the new analog circuits, especially in the case of analog schematic synthesis, such a very good thumb of rules can be dug out from the companion circuits, which were drawn before by the professional designers in hand. Also, such analog-aware schematic synthesis with companion circuits is very useful to analog migration between different technologies, which is very common in analog design due to the integrated-circuit technology progress.
Analog-aware schematic synthesis with companion circuits accepts the new circuit netlist, and the companion circuit schematic, mainly goes through such three steps: rule extraction from companion circuit schematic, rule extraction for new circuit, and rule application for new circuit schematic synthesis, and output the new circuit schematic in last, as shown in (a) of Fig. 55.

7.1 Rule extraction from companion circuit schematic

Rule extraction from companion circuits accepts the companion circuit schematic, mainly goes through the five steps: pre-processing, tracing direct current paths, tracing signal flow paths, exploring structural features, and exploring schematic rules from companion circuit schematic, and outputs the schematic rules for companion circuits, as shown in (b) of Fig. 55.
To leverage the schematic rules for new circuit as possible, rule extraction from companion circuits should cover group device level, direct current path level, block level, and more high level.

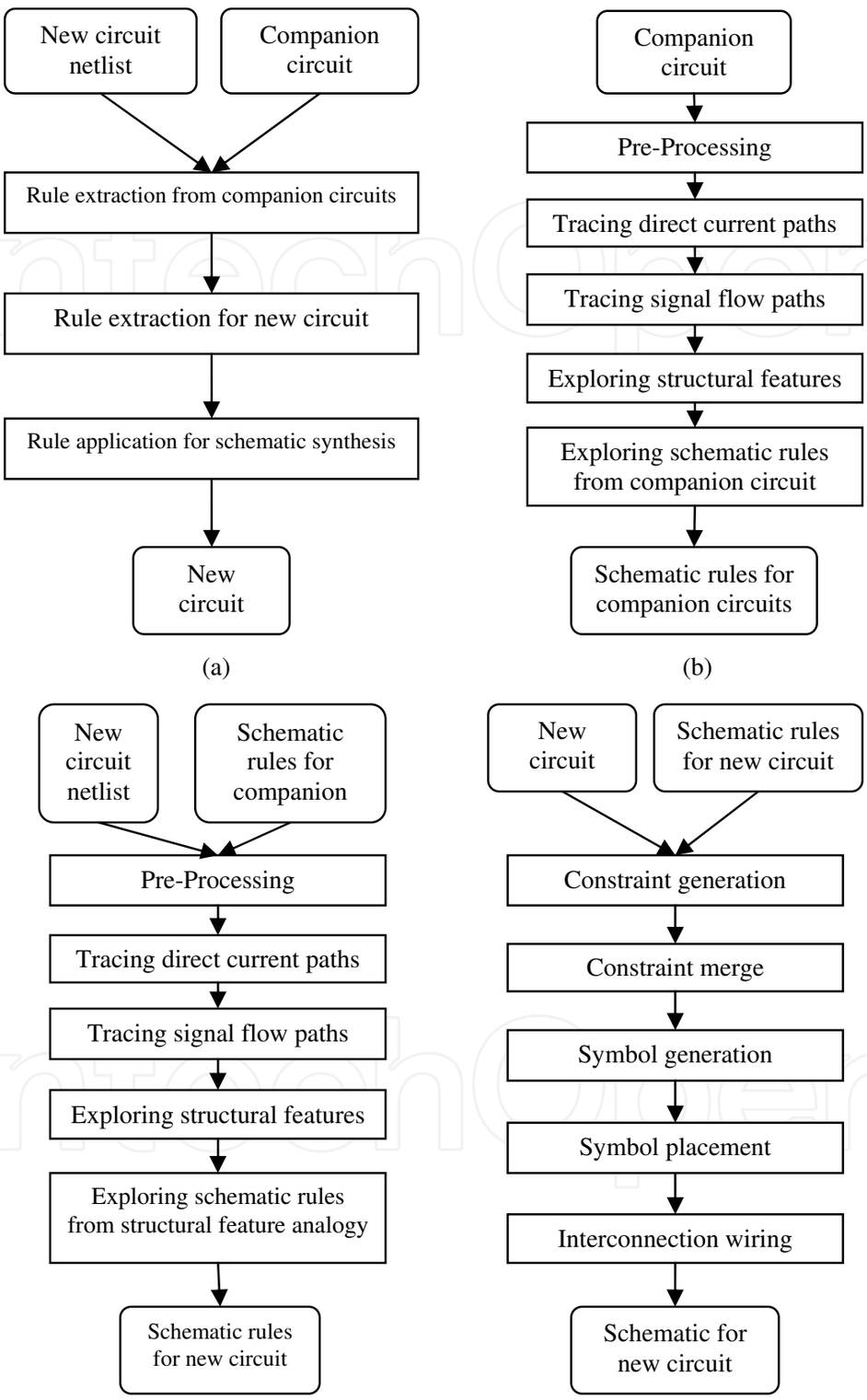


Fig. 55. Analog-aware schematic synthesis with companion circuits

7.2 Rule extraction for new circuit

Rule extraction from new circuits accepts the schematic rules from companion circuits and new circuit netlist, mainly goes through the five steps: pre-processing, tracing direct current paths, tracing signal flow paths, exploring structural features, and exploring schematic rules from structural feature analogy, and outputs the schematic rules for new circuits, as shown in (c) of Fig 55.

Most of the steps are same as previous descriptions except exploring schematic rules from structural feature analogy. Exploring schematic rules from structural feature analogy can be done on device level, direct current path branch level, direct current path level, block level and more high level, and in procedure the exploration should be started from low level structural feature comparison to high level structure feature comparison.

If a group of devices in new circuit has the same structural feature as a group of devices in companion circuits, the schematic rules for the group of devices in companion circuits will be copied for the group of devices in the new circuit.

If a direct current path in new circuit has the same structural feature as a direct current path in companion circuits, the schematic rules for the direct current path in companion circuits will be copied for direct current path in the new circuit.

If a block in new circuit has the same structural feature as a block in companion circuits, the schematic rules for the block in companion circuits will be copied for block in the new circuit.

If a new circuit has the same structural feature as a companion circuit, the schematic rules for the companion circuit will be copied for the new circuit.

7.3 Rule application for new circuit schematic synthesis

Rule application for new circuit schematic synthesis accepts the net circuit netlist and the schematic rules for new circuit, mainly goes through the five steps: constraint generation, merge constraints with schematic rules, symbol generation, symbol placement, and interconnection wiring, and outputs the schematic for new circuits, as shown in (d) of Fig 55.

Symbol generation includes the shape of symbols and the side location and side sequence for each terminal pin-out, which should refer that of companion circuits if the identical structural feature is found from the companion circuits, so the program needs to make a comparison for checking out the functional matching relations for circuits and the corresponding relation for terminal-to-terminal between new circuit and companion circuit.

The symbol placement includes the relative position, mirroring, rotating, symmetry, and alignment rules, which should refer that of companion circuits if the identical structural feature is found from the companion circuits, so the program needs to make a comparison for checking out the functional matching relations for circuits and the corresponding relation for device-to-device and block-to-block between new circuit and companion circuit.

The interconnection wiring includes the net self-symmetry, the net pair symmetry, and quasi-bus wiring, which should refer that of companion circuits if the identical structural feature is found from the companion circuits, so the program needs to make a comparison for checking out the functional matching relations for circuits and the corresponding relation for net-to-net between new circuit and companion circuit.

8. Experiments

We test the analog circuit schematic synthesis method with a flattened DAC circuit. After the functionality analysis and partitioning, new hierarchy is re-constructed; the constraints

for schematic generation, circuit and layout optimization are generated; and also the schematics are generated from the new hierarchy design, port types, and constraints. Part of the hierarchical design schematic is shown as in Fig. 56 – Fig. 59; the analog structural features can be got from the schematics intuitively.

The top circuit schematic is shown in Fig. 56, the top circuit is a digit-to-analog converter circuit, which consists of two op-amp circuits, one band-gap circuit, one bias circuit, and one DAC-core circuit. In this schematic, good layout symbols are generated, especially for op-amp, and the symbol placement follows the signal flow clearly, which gives an intuitive requirement on future floor-planning.

The DC-core circuit schematic is shown in Fig. 56, where the devices in a DC path are placed from top to down; all the DC paths are aligned; T-ladder circuit can be captured intuitively; the power down circuit (two inverters) are shown clearly; and mos-cap devices can be got from the power line directly. All those give a better feeling for the requirements of device placement in layout stage.

The op-amp circuit schematic is shown as Fig. 58, where the symmetry for differential pair devices, load devices, and tail current devices (self-symmetry) is reflected correctly; DC paths are also shown clearly and DC paths are placed with signal flow followed. All those give a better feeling for the requirements on symmetry, dc connection wiring minimization, signal wiring minimization, and necessary protections of the op-amp circuit in layout stage.

The band-gap circuit schematic is shown in Fig. 59, where the devices in a DC path are placed from top to down; the quasi-symmetry between two band-gap branches is followed; the power-down control logic circuits (two inverters) can be got from the schematic clearly; and the power-connected mos-cap devices and the ground-connected mos-cap devices can be got from the power line and ground line directly.

For clearness on circuit schematic, part of the constraints is not displayed, and due to the page number limitation, the non-analog-aware circuit schematic generation results from NLview and Cadence for this test case is not presented here, no any analog functionality are reflected there correctly.

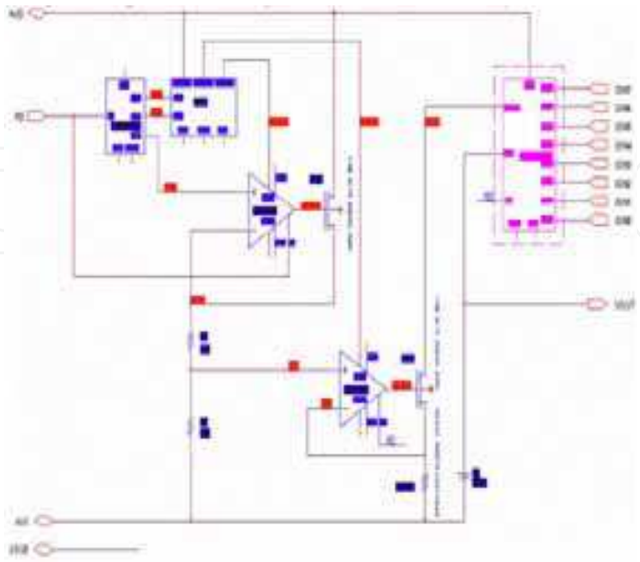


Fig. 56. Schematic of DAC

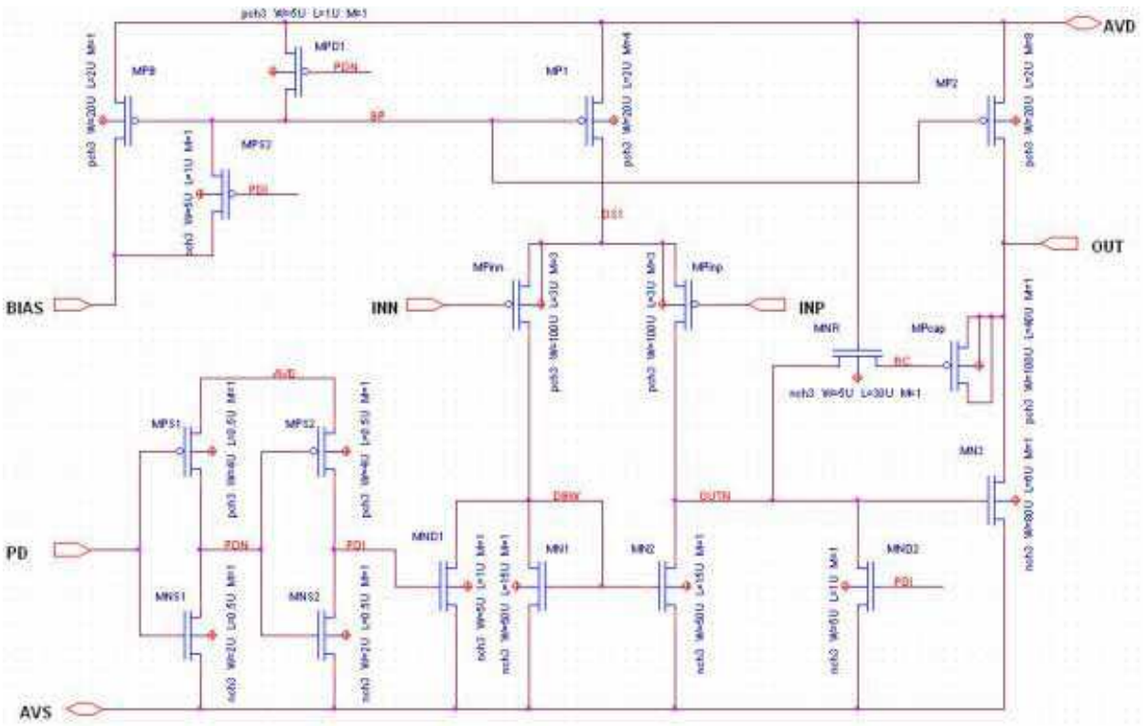


Fig. 57. Schematic of OPAMP

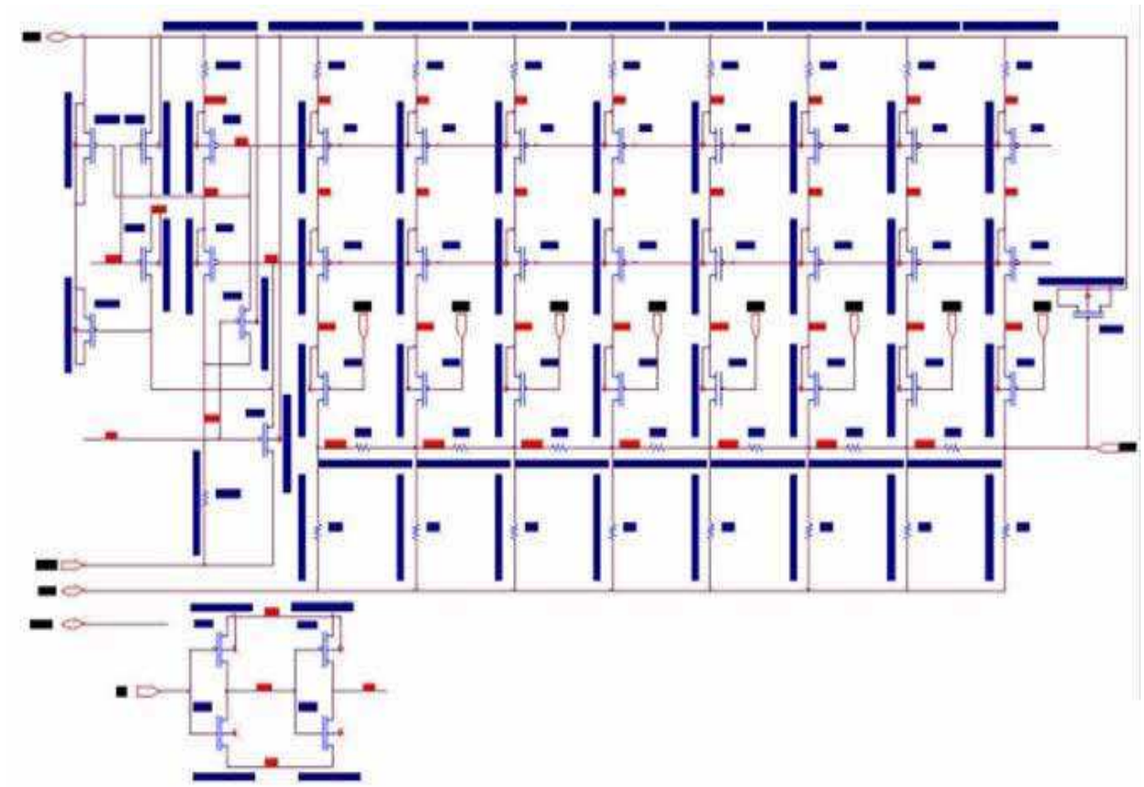


Fig. 58. Schematic of DAC-core

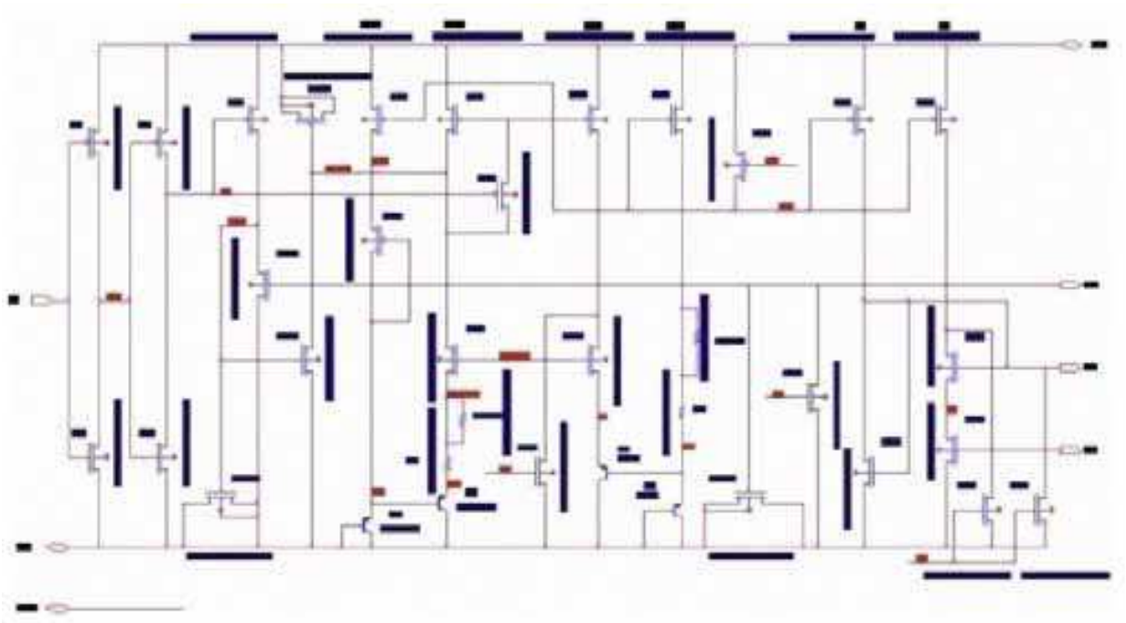


Fig. 59. Schematic of BANDGAP

9. Summary

Functionality analysis and partitioning technique can determine the functionality of analog design accurately and partition it into functionality-based hierarchy; further template based constraint generation can produce the constraints for schematic synthesis, circuit sizing, floor-planning, and layout optimization. With leverage of them, a novel analog schematic synthesis flow can produce analog-aware circuit schematics with functionality and structural features highlighted, also analog constraints are identified on schematic for circuit sizing, floor-planning, and layout optimization, which can be work as one of the base of analog synthesis to bridge topology synthesis and synthesis of circuit, floor-planning, and layout.

10. Reference

[1] Paul R. Gray, et al, "Analysis and design of analog integrated circuits", 4th edition, 2001.
[2] Behzad Razavi, "Design of analog CMOS integrated circuits", 2001.
[3] Phillip Allen, "CMOS analog circuit design", 2nd edition, 2002.
[4] Bemardinis, F.; Sangiovanni Vincentelli, A.; "Efficient analog platform characterization through analog constraint graphs", *IEEE ICCAD-2005,* pp415-421, Nov. 2005
[5] Malavasi, E.; Charbon, E.; Felt, E.; Sangiovanni-Vincentelli, A.; "Automation of IC layout with analog constraints", *IEEE Trans. On CAD,* vol. 15, no. 8, pp923 - 942, Aug. 1996
[6] Yiu-Cheong Tam; Young, E.F.Y.; Chu, C.; "Analog Placement with Symmetry and Other Placement Constraints", *IEEE ICCAD-2006,* pp349 - 354, Nov. 2006
[7] Jiayi Liu; Sheqin Dong; Xianlong Hong; Yibo Wang; Ou He; Goto, S.,"Symmetry constraint based on mismatch analysis for analog layout in SOl technology", ASP-DAC 2008, pp772 - 775, Mar. 2008

[8] Concept Engineering, "Nlview' Widgets: Customizable Schematic Generation Engines for EDA Tools"

[9] Wei-Ting Chen, Wen-Tsong Shiue, "Circuit schematic generation and optimization in VLSI circuits", *The Proceedings of IEEE Asia-Pacific Conference on Circuits and Systems 2004,* vol. 1, pp553 - 556, Dec. 2004

[10] Yuping Wu, "Research Reports on Analog Synthesis", *unpublished.*

[11] Graeb, H.; Zizala, S.; Eckmueller, J.; Antreich, K. "The sizing rules method for analog integrated circuit design" , IEEE/ACM International Conference on ICCAD 2001 , pp 343 – 349, 2001.

[12] Yuping Wu, "Novel method of analog circuit schematic synthesis", IEEE 8th International Conference on ASIC, pp1209-1212, 2009.

[13] Massier, T.; Graeb, H.; Schlichtmann, U.. "The Sizing Rules Method for CMOS and Bipolar Analog Integrated Circuit Synthesis", IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Volume: 27, Issue: 12, pp2209 – 2222, 2008.

[14] Pengfei Zhang, Xisheng Zhang, and Yuping Wu, "Signal flow driven circuit analysis and partitioning technique", United States Patent 7448003.

[15] Balasa, F.; Maruvada, S.C.; Krishnamoorthy, K.; "On the exploration of the solution space in analog placement with symmetry constraints", Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on Volume 23, Issue 2, Feb. 2004 Page(s):177 - 191

[16] Koda, S.; Kodama, C.; Fujiyoshi, K.; "Linear Programming-Based Cell Placement With Symmetry Constraints for Analog IC Layout", Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on Volume 26, Issue 4, April 2007 Page(s):659 - 668

[17] Changxu Du; Yici Cai; Xianlong Hong; Qiang Zhou; "A shortest-path-search algorithm with symmetric constraints for analog circuit routing", ASIC, 2005. ASICON 2005. 6th International Conference On Volume 2, 24-0 Oct. 2005 Page(s):844 - 847

[18] Qiang Ma,; Young, Evangeline F. Y.; Pun, K. P.; "Analog placement with common centroid constraints", Computer-Aided Design, 2007. ICCAD 2007. IEEE/ACM International Conference on 4-8 Nov. 2007 Page(s):579 - 585

[19] Koca, O.; Karl, H.; Weigel, R.; "A Novel Method Based Upon Nonlinear Optimization for Analog Filter Design with Mask Constraints"; Signals, Systems and Electronics, 2007. ISSSE '07. International Symposium on July 30 2007-Aug. 2 2007 Page(s):9 - 12

[20] Koca, O.; Karl, H.; Weigel, R.; "A New Approach for Analog Filter Design with Mask Constraints Utilizing Linear Programming"; Signals, Systems and Electronics, 2007. ISSSE '07. International Symposium on July 30 2007-Aug. 2 2007 Page(s):5 - 8

[21] Dhanwada, N.R.; Nunez-Aldana, A.; Vemuri, R.; "Component characterization and constraint transformation based on directed intervals for analog synthesis", VLSI Design, 1999. Proceedings. Twelfth International Conference On 7-10 Jan. 1999 Page(s):589 - 596

[22] Schwencker, R.; Eckmueller, J.; Graeb, H.; Antreich, K.; "Automating the sizing of analog CMOS circuits by consideration of structural constraints", Design, Automation and Test in Europe Conference and Exhibition 1999. Proceedings 9-12 March 1999 Page(s):323 - 327

[23] Yiu-Cheong Tam; Young, E.F.Y.; Chu, C.; "Analog Placement with Symmetry and Other Placement Constraints", Computer-Aided Design, 2006. ICCAD '06. IEEE/ACM International Conference on 5-9 Nov. 2006 Page(s):349 - 354

[24] Naiknaware, R.; Fiez, T.; "CMOS analog circuit stack generation with matching constraints", Computer-Aided Design, 1998. ICCAD 98. Digest of Technical Papers. 1998 IEEE/ACM International Conference on 8-12 Nov 1998 Page(s):371 - 375

[25] Mogaki, M.; Kato, N.; Shimada, N.; Yamada, Y.; "A layout improvement method based on constraint propagation for analog LSI's", Design Automation Conference, 1991. 28th ACM/IEEE June 17-21, 1991 Page(s):510 – 513.

[26] Donzelle, L.-O.; Dubois, P.-F.; Hennion, B.; Parissis, J.; Senn, P.; "A constraint based approach to automatic design of analog cells", Design Automation Conference, 1991. 28th ACM/IEEE June 17-21, 1991 Page(s):506 - 509

[27] Felt, E.; Charbon, E.; Malavasi, E.; Sangiovanni-Vincentelli, A.; "An efficient methodology for symbolic compaction of analog ICs with multiple symmetry constraints", Design Automation Conference, 1992. EURO-VHDL '92, EURO-DAC '92. European 7-10 Sept. 1992 Page(s):148 - 153

[28] Malavasi, E.; Charbon, E.; Felt, E.; Sangiovanni-Vincentelli, A., "Automation of IC layout with analog constraints", Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on Volume 15, Issue 8, Aug. 1996 Page(s):923 - 942

[29] De Bernardinis, F.; Sangiovanni Vincentelli, A.; "Efficient analog platform characterization through analog constraint graphs", Computer-Aided Design, 2005. ICCAD-2005. IEEE/ACM International Conference on 6-10 Nov. 2005 Page(s):415 - 421

[30] Fernanda Gusmão de Lima, Marcelo de O. Johann, José Luís Güntzel, Luigi Carro, Ricardo Reis, "A tool for analysis of universal logic gates functionality", Integrated Circuits and Systems Design, 1999. Proceedings. XII Symposium on 29 Sept.-2 Oct. 1999 Page(s):184 – 187

[31] Choudhury, U.; Sangiovanni-Vincentelli, A.; "Automatic generation of parasitic constraints for performance-constrained physical design of analog circuits", Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on Volume 12, Issue 2, Feb. 1993 Page(s):208 – 224.

[32] Zhe Zhou; Sheqin Dong; Xianlong Hong; Qingsheng Hao; Song Chen; "Analog constraints extraction based on the signal flow analysis"; ASIC, 2005. ASICON 2005. 6th International Conference On Volume 2, 24-0 Oct. 2005 Page(s):825 - 828

[33] Jiayi Liu; Sheqin Dong; Fei Chen; Xianlong Hong; Yuchun Ma; Di Long; "Symmetry Constraint for Analog Layout with CBL Representation", Solid-State and Integrated Circuit Technology, 2006. ICSICT '06. 8th International Conference on 23-26 Oct. 2006 Page(s):1760 - 1762

[34] Dhanwada, N.R.; Nunez-Aldana, A.; Vemuri, R.; "Hierarchical constraint transformation using directed interval search for analog system synthesis", Design, Automation and Test in Europe Conference and Exhibition 1999. Proceedings 9-12 March 1999 Page(s):328 - 335

[35] Choudhury, U.; Sangiovanni-Vincentelli, A.; "Constraint generation for routing analog circuits", Design Automation Conference, 1990. Proceedings., 27th ACM/IEEE 24-28 June 1990 Page(s):561 - 566

[36] Charbon, E.; Malavasi, E.; Sangiovanni-Vincentelli, A.; "Generalized constraint generation for analog circuit design", Computer-Aided Design, 1993. ICCAD-93. Digest of Technical Papers., 1993 IEEE/ACM International Conference on 7-11 Nov. 1993 Page(s):408 - 414

[37] Zhe Zhou; Sheqin Dong; Xianlong Hong; Qingsheng Hao; Song Chen, " Analog constraints extraction based on the signal flow analysis"; ASIC, 2005. ASICON 2005. 6th International Conference On Volume 2, 24-0 Oct. 2005 Page(s):825 – 828

[38] Kumar Arya, Swaminathan Misra, "Automatic Generation of Digital System Schematic Diagrams", Design Automation 1985. 22nd Conference on 23-26 June 1985 Page(s):388 – 395.

[39] Kumar Arya, Swaminathan Misra, "Automatic Generation of Digital System Schematic Diagrams" , Design & Test of Computers, IEEE Volume 3, Issue 1, Feb. 1986 Page(s):58 – 65.

[40] Swinkels, G.M.; Hafer, L , "Schematic generation with an expert system", Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on Volume 9, Issue 12, Dec. 1990 Page(s):1289 – 1306.

[41] Wei-Ting Chen, Wen-Tsong Shiue, "Circuit schematic generation and optimization in VLSI circuits", Circuits and Systems 2004 Proceedings. The 2004 IEEE Asia-Pacific Conference on Volume 1, 6-9 Dec. 2004 Page(s):553 - 556 vol.1.

[42] Kim, C.B., "Multiple mixed-level HDL generation from schematics for ASIC design", ASIC Conference and Exhibit, 1991. Proceedings Fourth Annual IEEE International 23-27 Sept. 1991 Page(s):P8 - 2/1-4.

[43] Tzi-Cker Chiueh , "HERESY: a hybrid approach to automatic schematic generation [for VLSI]", Design Automation. EDAC. Proceedings of the European Conference on 25-28 Feb. 1991 Page(s):419 – 423.

[44] Lee T.D., McNamee, L.P, "Structure optimization in logic schematic generation", Computer-Aided Design, 1989. ICCAD-89. Digest of Technical Papers, 1989 IEEE International Conference on 5-9 Nov. 1989 Page(s):330 – 333.

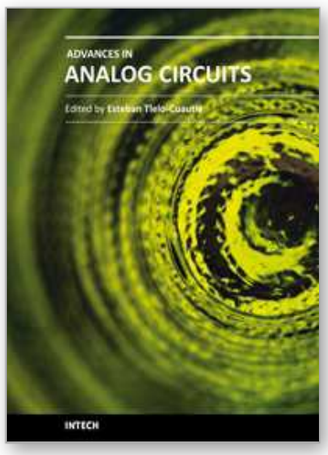
[45] Green Andersen, "Automated generation of analog schematic diagrams", Circuits and Systems, 1990, IEEE International Symposium on 1-3 May 1990 Page(s):3197 - 3200 vol.4.

[46] Zhan R., Feng H., Wu Q., Chen G., Guan X., Wang A.Z., "A new algorithm for ESD protection device extraction based on subgraph isomorphism", Circuits and Systems, 2002. APCCAS '02, 2002 Asia-Pacific Conference on Volume 2, 28-31 Oct. 2002 Page(s):361 - 366 vol.2

[47] J.R. Ullmann, "An Algorithm for Subgraph Isomorphism," J. Assoc. for Computing Machinery, vol. 23, pp. 31-42, 1976.

[48] Luigi P. Cordella, Pasquale Foggia, Carlo Sansone, and Mario Vento, "A (Sub)Graph Isomorphism Algorithm for Matching Large Graphs", IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, VOL. 26, NO. 10, OCTOBER 2004, Page(s):1367-1372.

[49] Bilal Radi A'Ggel Al-Zabi, Andriy Kernytskyy, Mykhaylo Lobur, Serhiy Tkatchenko, "On Graph Isomorphism Determining Problem", MEMSTECH'2008, May 21-24, 2008, Polyana, Page(s):84.



Advances in Analog Circuits

Edited by Prof. Esteban Tlelo-Cuautle

ISBN 978-953-307-323-1
Hard cover, 368 pages
Publisher InTech
Published online 02, February, 2011
Published in print edition February, 2011

This book highlights key design issues and challenges to guarantee the development of successful applications of analog circuits. Researchers around the world share acquired experience and insights to develop advances in analog circuit design, modeling and simulation. The key contributions of the sixteen chapters focus on recent advances in analog circuits to accomplish academic or industrial target specifications.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821

IntechOpen